Qiang Huo
Bin Ma
Eng-Siong Chng
Haizhou Li (Eds.)

LNAI 4274

Chinese Spoken Language Processing

5th International Symposium, ISCSLP 2006
Singapore, December 2006
Proceedings

Springer

Lecture Notes in Artificial Intelligence 4274

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Qiang Huo   Bin Ma
Eng-Siong Chng   Haizhou Li (Eds.)

Chinese Spoken Language Processing

5th International Symposium, ISCSLP 2006
Singapore, December 13-16, 2006
Proceedings

Springer

Volume Editors

Qiang Huo
The University of Hong Kong
Department of Computer Science
Hong Kong
E-mail: qhuo@cs.hku.hk

Bin Ma
Haizhou Li
Institute for Infocomm Research
21 Heng Mui Keng Terrace, 119613, Singapore
E-mail: {hli,mabin}@i2r.a-star.edu.sg

Eng-Siong Chng
Nanyang Technological University
School of Computer Engineering, Singapore
E-mail: aseschng@ntu.edu.sg

Library of Congress Control Number: 2006937161

CR Subject Classification (1998): I.2.7, F.4.2-3, I.2, H.3, I.7

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-49665-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-49665-6 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11939993 06/3142 5 4 3 2 1 0

Message from the General Chair

As the General Chair, I am indeed honored to pen the first words of the ISCSLP 2006 proceedings. On this occasion, I share with you the excitement in our community as we approach the end-point of an eventful, two-year journey, while anticipating a bright future, full of stimulating and inspiring scientific programs. Over the last two years, exciting research endeavors in Chinese spoken language processing have been pursued vigorously all over the world. I feel privileged to chair the Fifth ISCSLP in Singapore this year, particularly at this juncture in our history when the Chinese language is receiving worldwide attention, placing us at an important period of growth and change.

ISCSLP 2006 received a surprisingly large number of paper submissions. I would like to start by expressing my thanks to everyone who submitted their research work. This year's symposium was made possible by the hard work of many people. My deepest thanks to all the chairs, especially, the Technical Chairs, Qiang Huo and Bin Ma, together with the Special Session Chairs, who put an immense amount of quality work into the paper review and preparation of the technical program. I am also grateful to the Publication Chair, Eng-Siong Chng, who did a great job coordinating the publication of our conference proceedings (selected papers) in the Springer LNAI series for the first time.

I would also like to express my gratitude and appreciation to the Steering Committee, which has led ISCSLP to what it is today, for their invaluable advice at various stages of our development; to the members of the Organizing Committee, who made this event possible in Singapore; to the enthusiasts who contributed their ideas in the committee meetings in Lisbon, Toulouse and Pittsburgh; to all the sponsors for their generous support of our CSLP undertakings; and to the invited speakers as well as all the participants.

ISCSLP 2006 was unique in many ways, yet we continue the symposium's heritage of staying close to tradition. This year, we presented a scientific program of cutting-edge research in CSLP in the form of paper presentations, posters and demonstrations, supported by tutorials, workshops, and as always, good food. The program continues to show a healthy balance of a high standard of selection coupled with wide participation from the scientific community.

December 2006

Haizhou Li
General Chair, ISCSLP 2006

Preface

This book contains a selection of refereed papers presented at the Fifth International Symposium on Chinese Spoken Language Processing (ISCSLP 2006), held in Singapore during December 13-16, 2006. ISCSLP is a biennial conference for scientists, researchers, and practitioners to report and discuss the latest progress in all scientific and technological aspects of Chinese spoken language processing (CSLP). The previous four conferences were held in Singapore (ISCSLP 1998), Beijing (ISCSLP 2000), Taipei (ISCSLP 2002) and Hong Kong (ISCSLP 2004), respectively. Since its inception, ISCSLP has become the world's largest and most comprehensive technical conference focused on Chinese spoken language processing and its applications.

This year, a total number of 183 full paper submissions were received from 18 countries and regions, including mainland China, Taiwan, Hong Kong, Singapore, Japan, India, Korea, USA, Australia, Spain, Finland, Belgium, Czech Republic, Germany, Iran, Netherlands, Thailand, and Vietnam. Each paper was reviewed rigorously by at least two reviewers, thanks to the help offered by all members of the Technical Program Committee (TC), several members of the Organizing Committee (OC), and some additional reviewers. Detailed comments from the reviewers were given to authors for their consideration in revising the final manuscripts. Given the good quality of submitted papers, the TC and OC worked very hard to place as many of these good papers into the technical program as possible. As a result, 74 high-quality papers were selected to be included in this book. They are arranged in the following sections:

- Invited plenary talks and tutorials
- Topics in speech science
- Speech analysis
- Speech synthesis and generation
- Speech enhancement
- Acoustic modeling for automatic speech recognition
- Robust speech recognition
- Speech adaptation/normalization
- General topics in speech recognition
- Large vocabulary continuous speech recognition
- Multilingual recognition and identification
- Speaker recognition and characterization
- Spoken language understanding
- Human language acquisition, development and learning
- Spoken and multimodal dialog systems
- Speech data mining and document retrieval
- Machine translation of speech
- Spoken language resources and annotation

The conference program featured four invited plenary talks, two tutorials, four special sessions and a number of regular oral and poster sessions covering a wide range of areas related to CSLP. The program was further enhanced by a co-located event, “Affective Sciences Workshop,” jointly organized by the Swiss House Singapore, the Swiss Centre (NCCR) for Affective Sciences and ISCSLP 2006.

On behalf of the Organizing and Technical Program Committees, we would like to take this opportunity to express our deep gratitude to all the invited plenary speakers (Stephanie Seneff, Klaus R. Scherer, Franz Josef Och, and Tat-Seng Chua) and tutorial speakers (Keiichi Tokuda and Hang Li) for accepting our invitation to give talks at ISCSLP 2006. Helen Meng did an excellent job in organizing the plenary and tutorial sessions. Special thanks are due to the organizers of the special sessions (Chiu-Yu Tseng, Thomas Fang Zheng, Hsin-Min Wang, and Jianhua Tao), for their contributions in making ISCSLP 2006 an even more interesting and valuable event. We also want to thank those who helped review papers, and members of the International Advisory Committee for their advice. Last but not least, a special “thank you” has to go to the authors of papers and all the participants of the event for their support of ISCSLP.

We hope that the fruitful technical interactions made possible by this conference and the papers published in this book will benefit your research and development efforts in CSLP.

December 2006

Qiang Huo and Bin Ma
Technical Program Co-chairs
ISCSLP 2006

Eng-Siong Chng
Publication Chair
ISCSLP 2006

Organizing Committee

Honorary Chair

Susanto Rahardja	Institute for Infocomm Research, Singapore

General Chair

Haizhou Li	Institute for Infocomm Research, Singapore

Technical Program Chairs

Qiang Huo	The Univ. of Hong Kong, Hong Kong
Bin Ma	Institute for Infocomm Research, Singapore

Steering Committee Chair

Chin-Hui Lee	Georgia Institute of Technology, USA

General Secretary

Minghui Dong	Institute for Infocomm Research, Singapore

Plenary and Tutorial Chair

Helen Meng	The Chinese Univ. of Hong Kong, Hong Kong

Publication Chair

Eng-Siong Chng	Nanyang Technological Univ., Singapore

Publicity Chairs

Chung-Hsien Wu	National Cheng Kung Univ., Tainan
George M. White	Institute for Infocomm Research, Singapore

Special Session Chairs

Jianhua Tao	Chinese Academy of Sciences, Beijing
Chiu-Yu Tseng	Academia Sinica, Taipei
Hsin-Min Wang	Academia Sinica, Taipei
Thomas Fang Zheng	Tsinghua Univ., Beijing

Sponsorship Chair

Min Zhang	Institute for Infocomm Research, Singapore

Exhibition Chair

Yeow Kee Tan	Institute for Infocomm Research, Singapore

Local Arrangements Chair

Swee Lan See	Institute for Infocomm Research, Singapore

Technical Committee

Co-chairs

Qiang Huo	The Univ. of Hong Kong, Hong Kong
Bin Ma	Institute for Infocomm Research, Singapore

Members

Lian-Hong Cai	Tsinghua Univ., Beijing
C. F. Chan	City Univ. of Hong Kong, Hong Kong
Sin-Horng Chen	Chiao Tung Univ., Hsinchu
Yan-Ming Cheng	Motorola Labs, Schaumburg, IL, USA
Jen-Tzung Chien	National Cheng Kung Univ., Tainan
Lee-Feng Chien	Academic Sinica, Taipei
P. C. Ching	The Chinese Univ. of Hong Kong, Hong Kong
Wu Chou	Avaya Labs Research, USA
Min Chu	Microsoft Research Asia, Beijing
Jianwu Dang	JAIST, Japan
Li Deng	Microsoft Research, Redmond, USA
Li-Min Du	Chinese Academy of Sciences, Beijing
Qian-Jie Fu	House Ear Institute, USA
Pascale Fung	Hong Kong Univ. of Science and Technology, Hong Kong
Yuqing Gao	IBM T.J. Watson Research Center, Yorktown Heights, USA
Yifan Gong	Microsoft Corp., Redmond, USA
Mei-Yuh Hwang	Univ. of Washington, USA
Hui Jiang	York Univ., Canada
Chin-Hui Lee	Georgia Institute of Technology, USA
Lin-shan Lee	National Taiwan Univ., Taipei
Tan Lee	The Chinese Univ. of Hong Kong, Hong Kong
Shu-Hung Leung	City Univ. of Hong Kong, Hong Kong
Ai-Jun Li	Chinese Academy of Social Sciences, Beijing
Haizhou Li	Institute for Infocomm Research, Singapore
Peter Qi Li	Li Creative Technologies, USA
Jia Liu	Tsinghua Univ., Beijing
Yang Liu	Univ. of Texas at Dallas, USA
Kim-Teng Lua	COLIPS, Singapore
Xiao-Qiang Luo	IBM T.J. Watson Research Center, Yorktown Heights, USA
Brian Mak	Hong Kong Univ. of Science and Technology, Hong Kong

Man-Wai Mak	The Hong Kong Polytechnic Univ, Hong Kong
Helen Meng	The Chinese Univ. of Hong Kong, Hong Kong
Man-Hung Siu	Hong Kong Univ. of Science and Technology, Hong Kong
Frank Soong	Microsoft Research Asia, Beijing
Virach Sornlertlamvanich	Thai Computational Linguistics Laboratory (TCL), NICT, Thailand
Jianhua Tao	Chinese Academy of Sciences, Beijing
Chiu-Yu Tseng	Academia Sinica, Taipei
Haifeng Wang	Toshiba (China) Co., Ltd., Beijing
Hsiao-Chuan Wang	National Tsing Hua Univ., Hsinchu
Hsin-Min Wang	Academic Sinica, Taipei
Jhing-Fa Wang	National Cheng Kung Univ., Tainan
Kuansan Wang	Microsoft Research, Redmond, USA
Ye-Yi Wang	Microsoft Research, Redmond, USA
Chung-Hsien Wu	National Cheng Kung Univ., Tainan
Jim Wu,	Nuance Communication, USA
Bo Xu	Chinese Academy of Sciences, Beijing
Yonghong Yan	Chinese Academy of Sciences, Beijing
Eric Yun-Yang Zee	City Univ. of Hong Kong, Hong Kong
Yunxin Zhao	Univ. of Missouri - Columbia, USA
Thomas Fang Zheng	Tsinghua Univ., Beijing
Yiqing Zu	Motorola China Research Center, Shanghai

Additional Reviewers

Berlin Chen	National Taiwan Normal Univ., Taipei
Minghui Dong	Institute for Infocomm Research, Singapore
Pasi Franti	Univ. of Joensuu, Finland
Yu Hu	The Univ. of Hong Kong, Hong Kong
Xugang Lu	JAIST, Japan
Namunu Chinthake Maddage	Institute for Infocomm Research, Singapore
Jun Xu	Creative Technology, Singapore
Nengheng Zheng	The Chinese Univ. of Hong Kong, Hong Kong
Donglai Zhu	Institute for Infocomm Research, Singapore

International Advisory Committee

Sin-Horng Chen	Chiao Tung Univ., Hsinchu
P. C. Ching	The Chinese Univ. of Hong Kong, Hong Kong
Di-Tang Fang	Tsinghua Univ., Beijing
Hsiao-Wen Hon	Microsoft Research Asia, Beijing
Taiyi Huang	Chinese Academy of Sciences, Beijing
Xuedong Huang	Microsoft Corp., Redmond, USA
Biing Hwang Juang	Georgia Institute of Technology, USA
Chin-Hui Lee	Georgia Institute of Technology, USA
Lin-Shan Lee	National Taiwan Univ., Taipei
Mao-Can Lin	Chinese Academy of Social Sciences, Beijing
Kim-Teng Lua	COLIPS, Singapore
Shinan Lv	Chinese Academy of Sciences, Beijing
Hwee-Tou Ng	National Univ. of Singapore
Chiu-Yu Tseng	Academia Sinica, Taipei
Hsiao-Chuan Wang	Tsinghua Univ., Hsinchu
Jhing-Fa Wang	National Cheng Kung Univ., Tainan
Ren-Hua Wang	Univ. of Science and Technology of China, Hefei
William Shi-Yuan Wang	The Chinese Univ. of Hong Kong, Hong Kong
Zuo-Ying Wang	Tsinghua Univ., Beijing
Victor Zue	Massachusetts Institute of Technology, USA

Sponsors

Gold Sponsors

InCampus
Nokia
Toshiba (China) Research and Development Center
WholeTree Technologies

Sponsors

Asian Federation of Natural Language Processing
Chinese Corpus Consortium
Chinese and Oriental Language Information Processing Society
IEEE Singapore Section
Institute for Infocomm Research
International Speech Communication Association
School of Computer Engineering, Nanyang Technological University
School of Computing, National University of Singapore
Swiss House Singapore

Table of Contents

Plenary

Tutorial

Topics in Speech Science

Speech Analysis

Speech Synthesis and Generation

Speech Enhancement

Acoustic Modeling for Automatic Speech Recognition

Robust Speech Recognition

Speech Adaptation/Normalization

General Topics in Speech Recognition

Large Vocabulary Continuous Speech Recognition

Multilingual Recognition and Identification

Speaker Recognition and Characterization

Spoken Language Understanding

Human Language Acquisition, Development and Learning

Spoken and Multimodal Dialog Systems

Speech Data Mining and Document Retrieval

Machine Translation of Speech

Spoken Language Resources and Annotation

Interactive Computer Aids for Acquiring Proficiency in Mandarin

Stephanie Seneff

Computer Science and Artificial Intelligence Laboratory,
MIT, Cambridge, MA 02139, USA
seneff@csail.mit.edu

Abstract. It is widely recognized that one of the best ways to learn a foreign language is through spoken dialogue with a native speaker. However, this is not a practical method in the classroom due to the one-to-one student/teacher ratio it implies. A potential solution to this problem is to rely on computer spoken dialogue systems to role play a conversational partner. This paper describes several multilingual dialogue systems specifically designed to address this need. Students can engage in dialogue with the computer either over the telephone or through audio/typed input at a Web page. Several different domains are being developed, in which a student's conversational interaction is assisted by a software agent functioning as a "tutor" which can provide them with translation assistance at any time. Thus, two recognizers are running in parallel, one for English and one for Chinese. Some of the research issues surrounding high-quality spoken language translation and dialogue interaction with a non-native speaker are discussed.

1 Introduction

It is widely agreed among educators that the best way to learn to speak a foreign language is to engage in natural conversation with a native speaker of the language. Yet this is also one of the most costly ways to teach a language, due to the inherently one-to-one student-teacher ratio that it implies.

Mandarin Chinese is one of the most difficult languages for a native English speaker to learn. Chinese is substantially more difficult to master than the traditional European languages currently being taught in America – French, Spanish, German, etc., because of the lack of common roots in the vocabulary, the novel tonal and writing systems, and the distinctly different syntactic structure.

With the rapid recent emergence of China as a major player in the global economy, there is an increased urgency to find ways to accelerate the pace at which non-native speakers can acquire proficiency in communicating in Chinese. It is evident that, as China becomes internationalized, individuals who can speak Chinese fluently will have a distinct advantage in tapping into the human, financial, and physical resources that China offers to the world. China itself has wholeheartedly embraced the need for the members of its society to acquire fluency in English, but the Western nations have been slow to reciprocate. Part of the problem is the shortage of educators who speak both English and Chinese

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 1–12, 2006.

fluently (at least in the U.S.) and who understand the pedagogy of language teaching.

Computers can offer a solution to this problem, both by engaging the student in one-on-one spoken conversation, where the computer role plays the conversational partner, and by providing translation assistance when needed to help the student formulate their half of the conversation. Conversations will ultimately support a wide range of topics and will likely be goal directed, to help hold the student's interest and focus their attention. These conversations need not be speech only, but instead could incorporate a display component, ranging from an avatar to embody the voice to an entire video-game-like environment [13,14].

The explosive expansion of computer usage in households around the world in the last decade is rapidly morphing into the widespread adoption of computers and personal digital assistants (PDA's) as devices for access to remote computational and information resources. Computers, via Voice over IP (VOIP), are also beginning to replace the land line and cellular telephone systems as an alternative way for humans to remotely communicate among one another. *Computer Aided Language Learning* (CALL) systems will be able to take advantage of the widespread availability of high data rate communications networks to support easy accessability to systems operating at remote sites. The student can just enter a Web page, where they would be able to type or speak to the system, with the system responding through displays and synthetic speech, supported by multimodal WIMP-based interaction.

Clearly, for this vision to become a reality, a considerable amount of research is necessary. While significant progress has been made on human language technologies, it is not clear that the technology is sufficiently mature to succeed in enticing students of Chinese to play computer conversational games. At issue is the very hard problem of speech recognition not only for a non-native speaker, but also for a hesitant and disfluent speaker. Environmental issues are another risk factor, as students could be using whatever set-up they have at home, and the developer has no control over microphone quality or placement, or over environmental noise. The quality of the provided translations must be essentially perfect, and the dialogue interaction must be able to gracefully recover from digressions and misinformation due to unavoidable recognition errors. Any multimodal interactions need to be intuitive and easily integrated into the conversational thread. Finally, computers should also be able to analyze the recorded utterances of the conversation, and, in a subsequent interaction, critique selected production errors, involving aspects such as phonetic accuracy [4,21], tone production [22,17], lexical and grammar usage [18], and fluency [7].

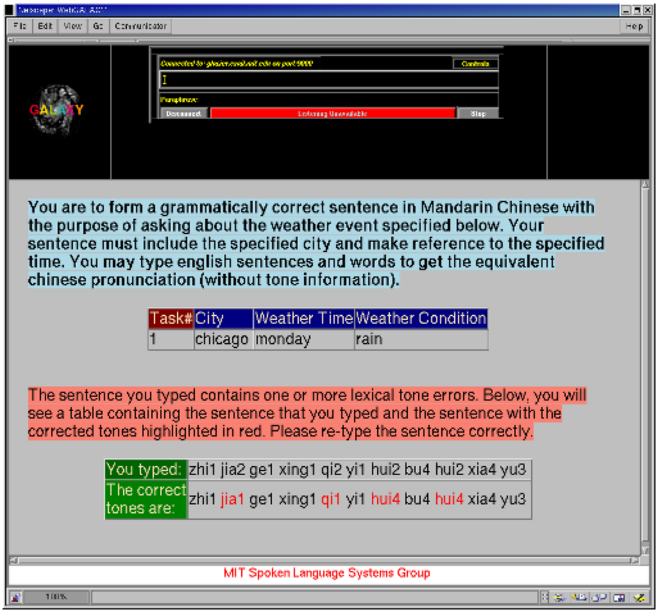
Holland et al. [8] have identified the basic principles of learning and cognition as (1) implicit feedback, (2) overlearning, and (3) adaptive sequencing. Implicit feedback falls out naturally in spoken conversational interaction – if the student does not speak fluently and with good articulation, the computer will not understand what they say. Furthermore, while the system could understand sentences that are slightly ill-formed, it could routinely paraphrase the student's query as a technique for both confirming understanding and providing implicit corrective feedback. Thus, the student might say, "yi1 *ge5* shu1" ("a book"), which the system would repair to "yi1 *ben3* shu1" in its paraphrase. Overlearning implies the achievement of a mastery of the material to the point of effortless and automatic retrieval. This can be achieved through the device used by video games to provide immediate feedback and intrinsic reward in the game itself. Adaptive sequencing involves careful design of the material to support incremental advances and to personalize the degree of difficulty to match the student's achievement level. Incremental advances in difficulty level will allow the student to be continuously challenged but not overchallenged. Computer language learning systems designed to achieve these goals will be entertaining and engaging, as well as educational.

At the Spoken Language Systems group in the Computer Science and Artificial Intelligence Laboratory at MIT, we have been developing multilingual spoken dialogue systems for nearly two decades [34]. A focus of our recent research has been to configure multilingual systems to support language learning applications [26]. Thus far, we have been concentrating on technology goals, but we hope to achieve a milestone of introducing the technology into the classroom wihtin the next year or so. Feedback from students and educators will lead to design changes which will eventually converge on a design that works best, given the constraints of the technology and the needs and interests of the students. Most especially, we hope to design application domains that will be entertaining to the students, thus engaging them in the activity and providing a rewarding and non-threatening learning experience.

2 Current Status

Our research on spoken conversational systems has focused on the travel domain: booking flights [29], city navigation, hotel booking, restaurant guide, weather information [33], etc. These topics are fortuitously often quite appropriate for the student of a second language, since it is likely that their first opportunity to utilize their language skills will be a visit to a country where the language is spoken. These systems center on goal-directed dialogue, which provides a focus for the conversation as well as an assessment mechanism based on task completion.

In developing these systems, we are attempting to provide generalizable technology solutions, especially for the linguistic analysis and the dialogue management strategy, which will lead to more rapid deployment of capabilities in other domains suitable for a language student. A recent new undertaking launched specifically for the language learning application is a kind of "symmetrical" dialogue interaction style, where the two dialogue participants jointly solve a shared problem, such as arranging a future meeting. Ultimately, we hope to empower language educators to design novel dialogue interaction scenarios on a wide range of topics, facilitated by an intuitive and easy-to-use graphical interface, modelled, for example, after MIT's *SpeechBuilder* system [9] or Carnegie Mellon's *Universal Speech Interface* [10].



You are to form a grammatically correct sentence in Mandarin Chinese with the purpose of asking about the weather event specified below. Your sentence must include the specified city and make reference to the specified time. You may type english sentences and words to get the equivalent chinese pronunciation (without tone information).

Task#	City	Weather Time	Weather Condition
1	chicago	monday	rain

The sentence you typed contains one or more lexical tone errors. Below, you will see a table containing the sentence that you typed and the sentence with the corrected tones highlighted in red. Please re-type the sentence correctly.

You typed:	zhi1 jia2 ge1 xing1 qi2 yi1 hui2 bu4 hui2 xia4 yu3
The correct tones are:	zhi1 jia1 ge1 xing1 qi1 yi1 hui4 bu4 hui4 xia4 yu3

Fig. 1. Screen shot of Web-based drill exercise, in which the student must solve a weather scenario. The student is provided explicit feedback on any tone errors.

To enable the student to gain competence in language usage within the scope of the exercise, we have developed a "translation game," where the system presents a word, phrase, or sentence in English, and the student is tasked with speaking an utterance with equivalent meaning in Chinese. If the computer judges their answer to be correct, it congratulates them and offers another (randomly generated) utterance to translate. The degree of difficulty of the translation task advances over time, and each student traverses this difficulty scale depending on their constantly monitored performance. A convenient parameter for measuring performance is the mean number of turns taken to successfully translate each posed utterance. An enrollment step allows the computer to personalize the level to the student's previously determined competence level across multiple episodic interactions.

Figure 1 shows an example of a text-based interface to a translation game. The system poses a simple scenario – Chicago; Monday; rain – and the student must formulate a query in pinyin that solves this scenario. The system can correct any tone errors, and also verifies if the student has correctly solved the scenario.

Having completed the translation exercises, the student would then attempt spoken dialogue interaction with the computer on a topic that exercises the same vocabulary and language constructs. The student converses (either by typing or by speaking) with a software agent that speaks only Chinese but has access to information sources. A software *tutor* can provide translation assistance at any time. The system automatically detects whether the student is speaking

English or Chinese – English utterances are translated whereas Chinese inputs are answered in Chinese. If the student doesn't understand the *response*, they can simply ask for a translation. To help reign in the language usage and thus improve the recognition performance, the tutor only provides translations that the Mandarin grammar can parse. The computer records a detailed log of the conversation, as well as capturing the student's spoken untterances as audio files that can be processed off-line for later language assessment.

An optional subsequent interaction provides the student with corrective fedeback on ways to improve their language production skills. While such feedback could be integrated into the original live conversation, we feel that it would be too distracting and disruptive during a time when they are concentrating on communicating their needs, and is thus best left as a follow-on drill exercise.

3 Related Research

In a 1998 review paper assessing the state of the art in computer aids for language teaching, Ehsani and Knodt [6] wrote: "Students' ability to engage in meaningful conversational interaction in the target language is considered an important, if not the most important, goal of second language education. This shift of emphasis has generated a growing need for instructional materials that provide an opportunity for controlled interactive speaking practice outside the classroom." However, perhaps because of the complex requirements associated with human-computer dialogue interaction, there has been surprisingly little research in spoken dialogue systems aimed towards this goal up to the present time.

There are a couple of promising ongoing initiatives, one in the U.S. and one in China, which are rapidly changing this picture. The U.S. initiative is the DARWARS Tactical Language Training System (TLTS) [13,14], which is part of the DARPA Training Superiority program. This ambitious program is targeted towards U.S. military personnel, and has focused thus far on Arabic as the target language. The idea is to embed language learning into a video-game-like environment, where the student assumes the role of a character in the video game, and interacts with other characters they encounter as they explore the virtual space. The student communicates with the other characters through speech and mouse-based gestures, and the options available at any point are based on the situational setting.

One of the presumably many ongoing efforts in China for learning English is the CSIEC Project [11,12], which is similar to ours in that the main delivery model is interactive dialogue at a Web page. Similar to the DARWARS project, the student interacts with embodied characters. No attempt is made to situate them in a complex scene, but rather each character simply role plays a conversational partner, mainly using a chatbot concept. The student can choose from among six different "virtual chatting partners," each of which has a distinct style of conversational interaction. For example, one personality will simply rephrase a user's statement into a question: "Why do you like to play baseball?" Another character will tell jokes or stories, or sing a song, upon request. Thus far interaction has been restricted to typed input, with the target language being English, although their intention is to eventually support spoken inputs. No translation assistance is offered.

Any *multilingual* spoken dialogue system could be relatively easily reconfigured as a language learning activity. For example, the ISIS system [20] is an impressive trilingual spoken dialogue system, supporting English, Mandarin, and Cantonese, which involves topics related to the stock domain and simulated personal portfolios.

A research topic that has some synergy with dialogue systems for language learning is the more general area of educational tutoring scenarios. An example involving spoken dialogue interaction to help a student solve simple physics problems can be found in [19].

4 Underlying Technologies

In this section, we describe the underlying technologies that support the language learning systems we are developing, highlighting three aspects in particular: (1) spoken language translation, (2) symmetrical dialogue interaction, and (3) assessment and feedback.

Our systems are all configured as a set of technology and interface servers that communicate among one another via a programmable central hub using the Galaxy Communicator architecture [25]. The student accesses the system simply by visiting a Web page. A Java audio program is automatically downloaded to support audio input at the computer. The audio stream is captured and transmitted to two speech recognizers at a remote server to allow the student to seamlessly switch between English and Chinese at any time. For speech recognition we use the SUMMIT landmark based system [5]. The natural language understanding component, TINA [27], receives a word graph of utterance hypotheses from both recognizers, and it is tasked both with selecting a candidate hypothesis and deciding which language was spoken. It also produces a *semantic frame*, encoding the meaning, which is then translated (paraphrased into Chinese by the language generation server), or answered (dispatched to the dialogue manager), if Chinese was spoken. The dialogue manager interprets the sentence in context, assisted by the context resolution server, and retrieves appropriate information pertinent to the question from the database (flights, weather, etc.). The dialogue manager prepares a *reply frame* which is passed on to the language generation server to produce a string response in Chinese. Each translation or response string is directed to the appropriate synthesizer (English or Chinese) by the hub program, and the response is played back to the student at the Web browser interface. When relevant, a separate HTML response is displayed as a table of appropriate information returned from the database. The system response is also displayed in a dialogue box that shows a sequence of all preceding user-system turns. The user's turn is represented by a paraphrase of the original user query (as understood by the system). This *paraphrase string* is automatically generated via formal rules by the language generation server.

At any time, the user can ask for a translation of the system's *response*, in which case the hub program redirects dialogue flow such that the previous reply frame is retrieved and generated in English instead of Chinese.

4.1 Speech Translation

One of the most challenging technology requirements is high quality translation of spoken inputs. This is a critical component of the system design, as it allows the student to dislodge from a situation where inadequate knowledge of the language stands in the way of advancing the dialogue. Two factors that make it feasible are (1) the student is speaking in their native language, and (2) the domain is highly restricted. Although statistical methods are currently dominating the field [1,16], linguistic methods are more likely to ultimately succeed in the special case of high quality spoken language translation within a narrow domain.

Our approach is based on the semantic frame playing the role of an interlingua. The natural language parser uses a language-specific grammar to transform the English or Chinese query into a common meaning representation, and language-specific generation rules are then applied to produce paraphrase strings in either language. This framework thus supports bidirectional translation as well as English-to-English and Chinese-to-Chinese paraphrases. Both the Chinese and English grammars are syntax-based. However, the terminals are lexicalized, and a spaciotemporal trigram language model, superimposed on the parse tree, provides significant constraint to aid in resolving parse ambiguity [27].

A trace mechanism to handle movement is important for maintaining consistency between the two meaning repesentations produced by the English and Chinese grammars respectively. English syntax moves wh-marked NP's to the front of the sentence, whereas in Chinese, temporals and locatives are typically topicalized in a similar fashion. The effort involved in porting to a new domain is minimized by relying heavily on syntactic rather than semantic structure.

Language generation makes use of the GENESIS [2] system. A set of recursive rule-templates operates top-down on the semantic frame, supported by a lexicon providing context-sensitive word senses. A preprocessor phase augments a possibly impoverished semantic frame with syntactic features appropriate for the target language [3], for example, supplying the inflectional endings for English verbs or the appropriate particle usage ("ge5," "ben3," "jian4," etc.) for quantified nouns in Chinese.

One advantage of a bidirectional linguistic-based translation method is that it provides a convenient mechanism for assessing the quality of the proposed translation. If a generated Chinese string fails to parse in the Chinese grammar, it is rejected by the system. In this way, the system never proposes a Chinese sentence that it can not understand.

To handle sentences for which direct translation fails to parse, we have developed two distinct back-off mechanisms, both of which are based on a simplified [*attribute: value*] representation of the meaning (which we refer to as an *"electronic form"* or *E*-form). The first one [30], utilized in the weather domain, exploits an example-based translation method, borrowing from ideas

Table 1. Spoken language translation results for English to Chinese, evaluated on an unseen test set for two domains. Recognition WER was 6.9% for the weather domain, and 10.6% for the flight domain. ORTH: text transcript; REC: recognizer output.

		Num Utts	Perfect	Acceptable	Incorrect	Yield	Accuracy
Weather	ORTH	695	613	43	9	95.7%	98.6%
	REC	695	577	50	13	92.1%	98.0%
Flights	ORTH	432	356	30	13	92.4%	96.7%
	REC	432	317	35	43	91.4%	89.1%

originating in the early '90's at ATR in Japan [24]. This method was found to be less effective in the flight domain, due to sparse data issues. Instead, we utilized an intermediate *English-to-English* paraphrase generated directly from the E-form, via relatively straightforward generation rules [32]. The English paraphrase is structured in a more regularized format than the original, and thus is more likely to yield a parsable translation.

We have evaluated our English-to-Chinese spoken language translation system in both the weather domain [30] and the flight domain [32], and results are summarized in Table 1. The 695 test utterances in the weather domain averaged 6.5 words per sentence, whereas the 432 test utterances in the flight domain averaged 5.6 words per utterance. Native speakers of Mandarin judged the translations as one of three categories: "Perfect," "Acceptable," and "Incorrect." The "yield" (percentage of utterances that were translated) and "accuracy" (percentage of utterances whose translation was judged to be acceptable or better) values reported in the table show that only the *accuracy* (89.1%) of the translations of the flight domain *recognition* outputs falls below 90%. It is often the case that an incorrect translation of a spoken input involves recognition errors that are easily communicated to the student through the provided English paraphrase, so that they would not be misinformed.

4.2 Symmetrical Conversational Paradigm

Typical first year language textbooks [31] center around the "self" – personal and family relationships, hobbies and personal preferences, and daily activities. Thus, we feel that it would be very useful if we could design generic dialogue systems that support a new style of interaction, involving a symmetrical or "equal party" conversational style. The participants share a common goal, but each side has only partial knowledge of the goal space. Thus new information is both solicited and provided by each participant in a conversational give-and-take paradigm.

We have chosen as our first scenario a simple game in which each participant role plays a persona with particular preferences with regard to activities (such as swimming, dancing, playing video games, etc.), along with a particular pre-arranged schedule of commitments over the next few days. The goal of the conversation is to find a time to get together to jointly participate in an activity that both parties find enjoyable. An example of a typical conversation in this domain is shown in Figure 2.

Student:	What are you doing tomorrow afternoon?
System:	I'm playing basketball. Would you like to join me?
Student:	No, I don't like to play basketball.
System:	What do you like to do?
Student:	I like to play video games and watch movies.
System:	Would you like to join me to watch a movie this evening?
Student:	No, I'm going to study this evening.
System:	Are you free tomorrow evening?
Student:	Yes, let's watch a movie tomorrow evening.

Fig. 2. Example English dialogue in the "hobbies and schedules" domain

The symmetry inherent in this kind of dialogue allows the computer to role play both sides of the conversation, thus enabling an effective resource for system development and refinement. A considerable amount of randomness can be built into the dialogue model, such that thousands of different variants can be automatically generated, which, incidentally, yield a large corpus of simulated utterances to train the initial statistics of the recognizer's language model, as well as providing example dialogues to show the student at a Web interface. Each time a new dialogue is generated, each participant's persona is reinstantiated to randomly selected values for preferences and schedule, and the initial question is randomly generated from among a number of different possibilities, e.g., "When are you free <day>?" What are you doing <day> <time>?" "Do you like <hobby>?" "What do you like to do?", etc.

Similar to our strategies in the past, we have separated out aspects particular to the scenario into an external domain-specific configuration file. This will make it relatively straightforward to redesign dialogue interaction scenarios around other topics that share a common abstraction with our original scenario, and thus lead to a substantially accelerated development cycle for an expanded curriculum. For example, a later scenario might involve making plans to dine together at a mutually agreeable restaurant. The dialogue manager adopts a simple E-form representation of its linguistic messages, which are converted into well-formed English and/or Chinese sentences using formal generation rules.

4.3 Assessment and Feedback

While we envision that students should not be distracted by *explicit* corrective feedback during their live conversation with the computer, implicit feedback in the form of (possibly corrected) paraphrases of their input sentence should allow them to learn to recognize and correct certain kinds of mistakes, such as particle usage mentioned above. We believe, furthermore, that students would welcome the opportunity to receive feedback in a follow-on exercise about any mistakes that were made and how they could be corrected. We are developing software aimed at detecting and repairing errors in tone production [22,23], grammar [18], and pronunciation [15].

Our work on tone production aims to provide the student with audio feedback that will easily draw their attention to the provided corrections. The student will be able to compare their original utterance with one in which the worst-scoring segments have been tonally adjusted. Since the two utterances differ only in the fundamental frequency dimension, it should be clear to the student what needs to be changed to correct the problem. This strategy requires a significant amount of signal processing. The first step is to automatically align the utterance with its orthographic transcript (this presumes that the transcript is correct). In parallel, the fundamental frequency contour is extracted. Next, an explicit parameterized model of the tone contour is computed for the vocalic portion of each syllable, and a score is calculated for the quality of the match to the expected tone contour for that syllable (having made adjustments for pitch range and for overall sentence declination effects on the fundamental frequency). Subsequently, for each syllable whose match is poor, the system can surgically repair the tone via phase vocoder techniques, reshaping it to match the predicted shape and height while preserving the original voice quality and speaker characteristics.

5 Summary and Future Work

This paper summarizes the current status of our research over the past several years, which is aimed towards providing an enriching, entertaining, and effective environment for practicing a foreign language by interacting with a computer. We are now at the threshold of a new phase of our research, in which we will introduce our technology into the classroom setting, and evaluate its effectiveness in teaching Mandarin to native speakers of English. We have barely begun the research on the assessment phase, which will involve post-processing the student's recorded utterances and providing focused corrective feedback on errors in prosodics, pronunciation, lexical, and grammar usage.

While our research has predominantly involved the paradigm of a native English speaker learning Mandarin, it would be quite straightforward to reverse the roles of the two languages to support a native Mandarin speaker learning English. Our attempts to support portability issues allow the techniques to generalize to other language pairs as well, but of course this needs to be demonstrated in future research.

Our symmetrical dialogue interaction paradigm could support the intriguing possibility of *humans* role playing both sides of the conversation, via their respective Web-based interfaces. Two students could interact with each other to solve the scenario, with the computer playing a tutorial role for both students, providing them with translation assistance when needed and filtering their utterances such that the other student only receives sentences spoken in Chinese.

In future research, we plan to greatly enrich the graphics component of our systems, ultimately supporting an interactive immersive video contextualization, thus blurring the boundary between educational exercises and video games.

Acknowledgements

Many former and current students and researchers in the Spoken Language Systems group at MIT have contributed to this research. They include Chih-Yu

Chao, Jonathan Lau, Vivienne Lee, John Lee, Mitch Peabody, Joe Polifroni, Julia Zhang, and, most especially, Chao Wang.

Julian Wheatley has been very supportive of our work and has helped us integrate it into the language laboratory at MIT. We are indebted to both students and educators at the Defense Language Institute, especially Mike Emonts, for their participation in pilot experiments introducing our software into the classroom.

This research was funded by grants from the Cambridge MIT Initiative, the Industrial Technology Research Institute, and the U.S. Department of Defense.

References

1. P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, R. L. Mercer, and P. S. Roossin, “A statistical approach to machine translation,” *Computational Linguistics,*, 16:2, 79–85, 1990.
2. L. Baptist and S. Seneff, “Genesis-II: A Versatile System for Language Generation in Conversational System Applications,” *Proc. ICSLP ’00*, V. III, 271–274, Beijing, China, Oct. 2000.
3. B. Cowan. “PLUTO: A Preprocessor for Multilingual Spoken Language Generation,” S.M. thesis, MIT Department of Electrical Engineering and Computer Science, February, 2004.
4. B. Dong, Q. Zhao, J. Zhang, and Y. Yan, “Automatic Assessment of Pronunciation Quality,” ISCSLP ’04, (2004) 137–140, Hong Kong.
5. J. Glass, “A Probabilistic Framework for Segment-Based Speech Recognition,” *Computer, Speech, and Language*, 17, 137-152, 2003.
6. F. Ehsani and E. Knodt, “Speech Technology in Computer-aided Language Learning: Strengths and Limitations of a new CALL Paradigm Language Learning & Technology,” V. 2, No. 1, 45-60, 1998.
7. M. Eskenazi, “Using Automatic Speech Processing for Foreign Language Pronunciation Tutoring: Some Issues and a Prototype,” Language Learning and Technolgy, V. 2, No. 2, 62-76, 1999.
8. V. M. Holland, J. D. Kaplan, and M. A. Sabol, “Preliminary Tests of Language Learning in a Speech-Interactive Graphics Microworld,” Calico Journal, V. 16, No. 3, 339-359, 1998.
9. *http://groups.csail.mit.edu/sls/technologies/speechbuilder.html*
10. *http://www.cs.cmu.edu/ usi/*
11. J. Jia, “The study of the application of a web-based chatbot system on the teaching of foreign languages,” *Proceedings of SITE 04*, AACE Press, USA. 2004. 1201-1207.
12. J. Jia, “CSIEC (Computer Simulator in Educational Communication): A Virtual Context-Adaptive Chatting Partner for Foreign Language Learners,” *Proceedings of ICALT 04*, IEEE Computer Society Press, USA. 2004. 690-692.
13. W. L. Johnson, S Marsella, N. Mote, M Si, H. Vihjalmsson, S. Wu, Balanced Perception and Action in the Tactical Language Training System, International Conference on Autonomous and Multi-agent Systems, 2004.
14. W. L. Johnson, S. Marsella, H. Vihjalmsson, “The DARWARS Tactical Language Training System,” *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)* 2004.
15. J-M Kim, C. Wang, M. Peabody, and S. Seneff, “An Interactive English Pronunciation Dictionary for Korean Learners,” *Interspeech ’04*, 1145-1148, 2004.

16. P. Koehn and F. J. Och and D. Marcu, “Statistical Phrase-Based Translation,” *Proc. HLT-NAACL*, 2003.
17. J. Leather, “Perceptual and productive learning of Chinese lexical tone by Dutch and English speakers.” In J. Leather and A. James, eds., *New Sounds 90*, University of Amsterdam, 72–97, 1990.
18. J.S.Y. Lee and S. Seneff, “Automatic Grammar Correction for Second-Language Learners,” *to appear*, *Proc. INTERSPEECH*, 2006.
19. D. Litman, “Spoken Dialogue for Intelligent Tutoring Systems; Opportunities and Challenges,” Keynote speech, *Proc. HLT-NAACL*, 2006.
20. H. Meng, P.C. Ching, S.F. Chan, Y.F. Wong, and C.C. Chan, “ISIS: An Adaptive Trilingual Conversational System with Interleaving, Interaction, and Delegation Dialogues,” *ACM Transactions on Computer-Human Interaction (TOCHI)*, V. 11, No. 3, pp 268–299, 2004.
21. A. Neri, C. Cucchiarini, and H. Strik, “Feedback in computer assisted pronunciation training: technology push or demand pull?” *Proceedings of ICSLP*, 1209–1212, 2002.
22. M. Peabody, S. Seneff, and C. Wang, “Mandarin Tone Acquisition through Typed Dialogues,” 173–176, *InSTIL Symposium on Computer Assisted Language Learning*, Venice, Italy, 2004.
23. M. Peabody and S. Seneff, “Towards Automatic Tone Correction in Non-native Mandarin,” *Submitted to ISCSLP ’06*, 2006.
24. S. Sato, “CTM: an example-based translation aid system using the character-based match retrieval method,” *Proc. COLING*, 1992.
25. S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, and V. Zue, “Galaxy-II: A Reference Architecture for Conversational System Development,” *ICSLP ’98*, 931–934, Sydney, Australia, December, 1998.
26. S. Seneff, C. Wang, M. Peabody, and V. Zue, “Second Language Acquisition through Human Computer Dialogue,” *Proc. ISCSLP ’04*, Hong Kong, 2004.
27. S. Seneff, “TINA: A Natural Language System for Spoken Language Applications,” *Computational Linguistics*, V. 18, No. 1, 61–86, 1992.
28. S. Seneff, C. Wang, and J. Zhang, “Spoken Conversational Interaction for Language Learning,” *Proc. InSTIL Symposium on Computer Assisted Language Learning*, 151–154, Venice, 2004.
29. S. Seneff, “Response Planning and Generation in the Mercury Flight Reservation System,” *Computer Speech and Language*, V. 16, 283–312, 2002.
30. C. Wang and S.Seneff, “High-quality Speech Translation for Language Learning,” 99–102, *InSTIL Symposium on Computer Assisted Language Learning*, Venice, Italy, 2004.
31. T.C. Yao and Y. Liu Yao, *Integrated Chinese, 2nd Edition*, *Cheng and Tsui Company*, Boston, MA, 2005.
32. S. Seneff, C. Wang, and J.S.Y. Lee, “Combining Linguistic and Statistical Methods for Bi-directional English Chinese Translation in the Flight Domain,” *To appear, AMTA ’06.*
33. V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T.J. Hazen, and L. Hetherington, “Jupiter: A Telephone-Based Conversational Interface for Weather Information,” *IEEE Trans. Speech and Audio Proc.,* 8(1), 85–96, 2000.
34. V. Zue and J. Glass, “Conversational Interfaces: Advances and Challenges,” *Proc. IEEE*, V. 88, No. 8, 1166–1180, 2000.

The Affective and Pragmatic Coding of Prosody

Klaus R. Scherer

Swiss Center for Affective Sciences
University of Geneva, Switzerland
klaus.scherer@pse.unige.ch

Abstract. Prosody or intonation is a prime carrier of affective information, a function that has often been neglected in speech research. Most work on prosody has been informed by linguistic models of sentence intonation that focus on accent structure and which are based on widely differing theoretical assumptions. Human speech production includes *both* prosodic coding of emotions, such as anger or happiness, *and* pragmatic intonations, such as interrogative or affirmative modes, as part of the language codes. The differentiation between these two types of prosody still presents a major problem to speech researchers. It is argued that this distinction becomes more feasible when it is acknowledged that these two types of prosody are differently affected by the so-called "push" and "pull" effects. Push effects, influenced by psychophysiological activation, strongly affect emotional prosody, whereas pull effects, influenced by cultural rules of expression, predominantly affect intonation or pragmatic prosody, even though both processes influence all prosodic production. The push-pull distinction implies that biological marking (push) is directly externalized in motor expression, whereas pull effects (based on socio-cultural norms or desirable, esteemed reference persons) will require the shaping of the expression to conform to these models. Given that the underlying biological processes are likely to be dependent on both the idiosyncratic nature of the individual and the specific nature of the situation, we would expect relatively strong inter-individual differences in the expressive patterns resulting from push effects. This is not the case for pull effects. Here, because of the very nature of the models that pull the expression, we would expect a very high degree of symbolization and conventionalization, in other words comparatively few and small individual differences. With respect to cross-cultural comparison, we would expect the opposite: very few differences between cultures for push effects, large differences for pull effects.

Scherer and collaborators have suggested that these two types of effect generate two general principles underlying the coding of emotional and pragmatic information in speech, *covariation* and *configuration*. The covariation principle assumes a continuous, but not necessarily linear, relationship between some aspect of the emotional response and a particular acoustic variable. Thus, if F0 is directly related to physiological arousal, F0 should be higher in rage as compared with mild irritation. An early study by Ladd and collaborators is described, showing that the F0 range shows a covariance relationship with attitudinal and affective information in that a larger range communicates more intense emotional meaning. In contrast, almost all linguistic descriptions assume that intonation involves a number of categorical distinctions, analogous to contrasts between segmental phonemes or between grammatical categories. In consequence, the configuration principle implies that the specific meaning conveyed by an utterance is actively inferred by the

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 13 – 14, 2006.

listener from the total prosodic configuration, such as "falling intonation contour", and the linguistic choices in the context.

The configuration principle seems to determine the coding of pragmatic features of speech, for example, emphasis, or message types such as declarative or interrogative mode. A study by Scherer and his collaborators is outlined that suggests that final pitch movements are coded by the configuration principle. While final-rise versus final-fall patterns of F0 in themselves do not carry emotional meanings; they are linked to sentence modes such as question versus affirmation. However, it can be shown that context, such as type of sentence, affects interpretation. Whereas a falling intonation contour is judged as neutral in a WH-question, it is judged as aggressive or challenging in a yes/no question. It can be hypothesized that continuous variables are linked to push effects (externalization of internal states); whereas configurations of category variables are more likely to be linked to pull effects (specific normative models for affect signals or display). In terms of origin and evolutionary development, it seems plausible to suggest that the covariation principle is evolutionarily continuous with the bio-psychological mechanism that underlies affect vocalizations in a large number of species. This possibility is described, for example, by the motivational-structural rules suggested by Morton in an attempt to systematize the role of fundamental frequency, energy, and quality (texture) of vocalization for the signaling of aggressive anger and fear. In contrast, the configuration principle might be assumed to be an evolutionarily more recent development, based on the emergence of language with its specific design features, including intonation patterns. Affective meaning could be produced by nonstandard usage of these respective codes, depending on the degree of context-dependent emotional marking. If this assumption is correct, one could imagine that the neural mechanisms that underlie the perceptual processing of the two types of affect messaging via prosodic variation are different, both with respect to the neural structures and circuits involved and to the nature and timing of the respective processes.

A preliminary step in the empirical testing of this prediction is an examination of the difference in neural auditory processing of speech samples communicating either emotional content (joy, anger, sadness) or linguistic-pragmatic meaning categories (e.g., statements or questions). If the prosodic communication of emotional content via the configuration principle uses a nonstandard, or marked, version of linguistic-pragmatic prosody identifying message type, it would be useful to first identify the potential neural processing differences between covariation-based emotion prosody patterns and linguistic-pragmatically coded prosodic message types. In this contribution, several empirical studies are described that exemplify the possibilities of dissociating emotional and linguistic prosody decoding at the behavioral and neurological level. The results highlight the importance of considering not only the distinction of different types of prosody, but also the relevance of the task accomplished by the participants, to better understand information processes related to human vocal expression at the suprasegmental level.

Challenges in Machine Translation

Franz Josef Och

Google Research
och@google.com

Abstract. In recent years there has been an enormous boom in MT research. There has been not only an increase in the number of research groups in the field and in the amount of funding, but there is now also optimism for the future of the field and for achieving even better quality. The major reason for this change has been a paradigm shift away from linguistic/rule-based methods towards empirical/data-driven methods in MT. This has been made possible by the availability of large amounts of training data and large computational resources. This paradigm shift towards empirical methods has fundamentally changed the way MT research is done. The field faces new challenges. For achieving optimal MT quality, we want to train models on as much data as possible, ideally language models trained on hundreds of billions of words and translation models trained on hundreds of millions of words. Doing that requires very large computational resources, a corresponding software infrastructure, and a focus on systems building and engineering. In addition to discussing those challenges in MT research, the talk will also give specific examples on how some of the data challenges are being dealt with at Google Research.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, p. 15, 2006.

Automatic Indexing and Retrieval of Large Broadcast News Video Collections - The TRECVID Experience

Tat-Seng Chua

School of Computing, National University of Singapore
chuats@comp.nus.edu.sg

Abstract. Most existing operational systems rely purely on automatic speech recognition (ASR) text as the basis for news video indexing and retrieval. While current research shows that ASR text has been the most influential component, results of large scale news video processing experiments indicate that the use of other modality features and external information sources such as the Web is essential in various situations. This talk reviews the frameworks and machine learning techniques used to fuse the ASR text with multi-modal and multi-source information to tackle the challenging problems of story segmentation, concept detection and retrieval in broadcast news video. This paper also points the way towards the development of scalable technology to process large news video archives.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, p. 16, 2006.

An HMM-Based Approach to Flexible Speech Synthesis

Keiichi Tokuda

Department of Computer Science and Engineering,
Nagoya Institute of Technology
tokuda@ics.nitech.ac.jp

Abstract. The increasing availability of large speech databases makes it possible to construct speech synthesis systems, which are referred to as corpus-based, data-driven, speaker-driven, or trainable approach, by applying statistical learning algorithms. These systems, which can be automatically trained, not only generate natural and high quality synthetic speech but also can reproduce voice characteristics of the original speaker. This talk presents one of these approaches, HMM-based speech synthesis. The basic idea of the approach is very simple: just train HMMs (hidden Markov models) and generate speech directly from them. To realize such a speech synthesis system, however, we need some tricks: algorithms for speech parameter generation from HMMs, and a mel-cepstrum based vocoding technique are reviewed, and an approach to simultaneous modeling of phonetic and prosodic parameters (spectrum, F0, and duration) is also presented. The main feature of the system is the use of dynamic feature: by inclusion of dynamic coefficients in the feature vector, the speech parameter sequence generated in synthesis is constrained to be realistic, as defined by the parameters of the HMMs. The attraction of this approach is that voice characteristics of synthesized speech can easily be changed by transforming HMM parameters. Actually, it has been shown that we can change voice characteristics of synthetic speech by applying a speaker adaptation technique which has been used in speech recognition systems. The relationship between the HMM-based approach and other concatenative speech synthesis approaches is also discussed. In the talk, not only the technical description but also recent results and demos will be presented.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, p. 17, 2006.

Text Information Extraction and Retrieval

Hang Li

Microsoft Research Asia
hangli@microsoft.com

Abstract. Every day people spend much time on creating, processing, and accessing information. In fact, most of the information exists in the form of "text", contained in books, emails, web pages, news paper articles, blogs, and reports. How to help people quickly find information from text data and how to help people discover new knowledge from text data has become an enormously important issue. Many research efforts have been made on text information extraction, retrieval, and mining; and significant progress has made in recent years. A large number of new methods have been proposed, and many systems have been developed and put into practical uses. This tutorial is aimed at giving an overview on two central topics of the area: namely Information Extraction (IE) and Information Retrieval (IR). Important technologies on them will be introduced. Specifically, models for IE such as Maximum Entropy Markov Model and Conditional Random Fields will be explained. Models for IR such as Language Model and Learning to Rank will be described. A brief survey on recent work on both IE and IR will be given. Finally, some recent work on the combined uses of IE and IR technologies will also be introduced.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, p. 18, 2006.

Mechanisms of Question Intonation in Mandarin

Jiahong Yuan

Department of Linguistics, University of Pennsylvania
Philadelphia, PA 19104, USA
jiahong@ling.upenn.edu

Abstract. This study investigates mechanisms of question intonation in Mandarin Chinese. Three mechanisms of question intonation have been proposed: an overall higher phrase curve, higher strengths of sentence final tones, and a tone-dependent mechanism that flattens the falling slope of the final falling tone and steepens the rising slope of the final rising tone. The phrase curve and strength mechanisms were revealed by a computational modeling study and verified by the acoustic analyses as well as the perception experiments. The tone-dependent mechanism was suggested by a result from the perceptual study: question intonation is easier to identify if the sentence-final tone is falling whereas it is harder to identify if the sentence-final tone is rising, and was revealed by the acoustic analyses on the final Tone2 and Tone4.

Keywords: question intonation, perception, acoustics.

1 Introduction

This study investigates mechanisms of question intonation in Mandarin Chinese. In terms of surface F0, the difference between question and statement intonation in Mandarin Chinese is quite diverse, which can be illustrated by Figure 1.

In general, the difference can be realized as the following: 1. The question curve is higher than the statement curve on the whole sentence (the left pair in Figure 1). 2. The question curve diverges from the statement curve after a point (the middle pair in Figure 1). 3. The question curve is higher than the statement curve except at some portions or points (the right pair in Figure 1).

The difference between question and statement intonation has attracted much attention in Chinese intonation study. De Francis claims that the whole pitch level of the interrogative is higher than that of the declarative [1]. Disagreeing with De Francis, Tsao argues that the whole pitch level has no difference between the two intonation types and interrogative intonation in Chinese is ‘a matter of stress’ [2]. Gårding models Chinese intonation with ‘grids’, which qualitatively mark a time-varying pitch range. Lexical tones then fit into that range [3, 4]. In Gårding’s model, the two intonation types have different grids. Shen J. proposes that the top line and the base line of a pitch contour are independent in the prosodic system of Chinese [5, 6]. For interrogative intonation the top line falls gradually whereas the base line undulates slightly and ends at a much higher point (compared to declarative intonation). Shen X. investigates the difference between the two intonation types by comparing their pitch values at four points: starting point, highest peak, lowest trough,

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 19 – 30, 2006.

and ending point [7]. Her conclusion is that interrogative intonation begins at a register higher than declarative, although it may end with either a high or low key. In Pan-Mandarin ToBI, question intonation is mainly associated with a high boundary tone in the intonational tones tier [8].

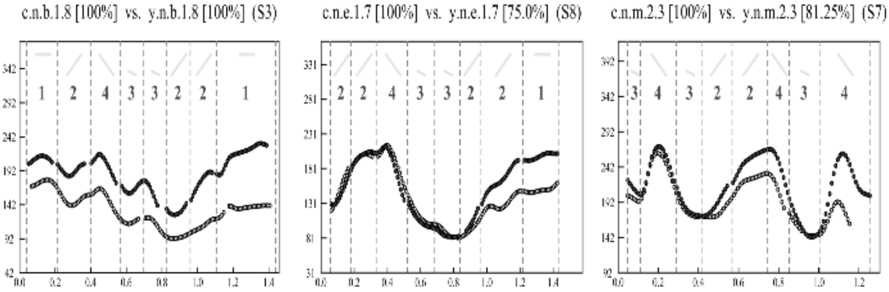


Fig. 1. Diverse patterns of difference between the question curve (dark circles) and the statement curve (open circles)

We studied the difference between question and statement intonation in Chinese with Stem-ML [9], an intonation description language combined with an algorithm for translating tags into quantitative prosody. Our study found that the ‘diverse’ difference between question and statement intonation in Mandarin Chinese can be accounted for by two consistent mechanisms: an overall higher phrase curve for the question intonation, and higher strength values of sentence final tones for the question intonation. It also suggested that the phrase curves of the two intonation types tend to be parallel and boundary tones are not necessary for modeling the difference between the two intonation types in Mandarin Chinese [10].

These results raised several interesting issues and questions: Firstly, they seem inconsistent with some previous models of Chinese intonation. For example, in Gårding’s model the two intonation types have grids with different directions and in Pan-Mandarin ToBI a high boundary tone has been used to transcribe question intonation. Secondly, if question intonation is realized by strengthening the final syllables then what is the difference between ‘plain’ interrogative intonation and interrogative intonation whose final syllable(s) are focused (given the fact that a focused syllable must also be strengthened in speech)? And what is the difference between ‘plain’ question intonation and declarative intonation that has a focus at its end? Thirdly and most importantly, do actual production and perception data support these results?

Perception and acoustic studies were then conducted in light of these issues and questions. A systematic corpus was created for both of these facets of the study. The corpus consisted of 130 sentences, which were minimal pairs contrasting on intonation type (statement, question), presence of a focus or not, focus position (sentence initial, middle, end), tone of the focused syllable (tone1, tone2, tone3, tone4), and tone of the last syllable (tone1, tone2, tone3, tone4). For example:

1. Li3bai4wu3 Luo2Yan4 yao4 mai3 mao1. ← statement, no focus
   Friday Luo2Yan4 will buy cat
   "Luo2Yan4 (Luo2 is a last name and Yan4 is a first name) will buy a cat Friday."
2. Li3bai4wu3 Luo2yan4 yao4 mai3 mao1? ← question, no focus
   Friday Luo2yan3 will buy cat
   "Luo2Yan4 will buy a cat Friday?" (It is a question with surprise, dubiousness, etc.)
3. Li3bai4wu3 Luo2Yan4 yao4 mai3 **mao1**. ← statement, a tone1 focus at the end
   Friday Luo2Yan4 will buy **cat**
   "(not a goat, not a deer,) LuoYan will buy a **cat** Friday."

Eight native Mandarin speakers, four male and four female, took part in the recording. Two perception experiments were conducted on the 1040 utterances recorded in the database. One was for identifying intonation type and the other for identifying focus. Sixteen listeners, 8 female and 8 male, participated in the perception experiments. The listeners are also native Mandarin speakers.

Results of the perception experiments have been reported in [11, 12]. In section 2 I summarize these results, and argue that the perception experiments found evidence for the strength mechanism of question intonation, as well as evidence for a third mechanism that was not found in our previous Stem-ML modeling study. In section 3, I report the results of acoustic analyses, which show further evidence for the strength and the phrase curve mechanisms, and discuss what the third mechanism is. Finally, in section 4, I present the conclusions.

2 Evidence from Perception Experiments

As reported in [11, 12], the perception experiments found that: 1. Statement intonation is easier to identify than question intonation; 2. The tone of the last syllable does not affect the identification of statement intonation; 3. The tone of the last syllable does affect the identification of question intonation: First, question intonation is easier to identify on a sentence ending with Tone4 than those ending with the other tones; second, identification of some speakers' question intonation is very difficult if the sentence ends with Tone2; 4. A focus at the end of a sentence makes statement intonation more difficult to identify; 5. A focus at the middle or the end of a sentence makes question intonation easier to identify; 6. A focus at the middle of a sentence is easier to identify than a focus at the beginning or at the end, no matter what intonation type the sentence has; 7. A focus at the end of a sentence is more difficult to identify than a focus at the beginning for a statement but not for a question; 8. The tone of a focused syllable does not affect focus identification under Statement, even if the focus is at the end of a sentence; 9. A focus on Tone2 is more difficult to identify than that on Tone4 if the focus is at the middle of a question sentence and a focus on Tone1 is more difficult to identify than that on Tone3 if it is at the end of a question sentence. Some of the results were also found in Liu and Xu's study on a different dataset [13].

These results reveal four interesting asymmetries: statement and question intonation identification; effects of the final tone2 and tone4 on question intonation identification; effects of the final focus on statement and question intonation identification; and effects of intonation type on focus identification.

The asymmetry of statement and question intonation identification manifests in two ways: First, statement intonation is easier to identify than question intonation, suggest by both a higher mean identification ratio and a smaller variation; second, the tone of the last syllable does not affect statement intonation identification but it does affect question intonation identification. The intonation identification test was a forced choice test: the listeners must identify the intonation type of each utterance as either a statement or a question. That question intonation identification was less accurate means that many question intonation utterances were identified as statements. This suggests that statement intonation is a default or unmarked intonation type. That is, listeners fall back to this option when there is not enough information suggesting 'question', which is also supported by the fact that the tone of the last syllable does not affect Statement identification. Question intonation is, however, a marked intonation type. It can only be identified if the listeners actually hear the 'question' features/mechanisms.

The second asymmetry revealed by the perception experiments is of the effects of the sentence-final Tone2 and Tone4 on question intonation identification: On the one hand, question intonation is easier to identify on a sentence ending with Tone4 than ending with the other tones. On the other hand, identification of some speakers' question intonation is very difficult if the sentence ends with Tone2. Tone4 is a falling tone and Tone2 is a rising tone. Therefore the asymmetry can also be stated as follows: In sentence-final position, question intonation is easier to identify on a falling tone and more difficult to identify on a rising tone. Our previous Stem-ML modeling study on Mandarin intonation showed that the difference between statement and question intonation in Mandarin Chinese can be accounted for by two mechanisms: an overall higher phrase curve for the question intonation, and higher strength values of sentence final tones for the question intonation. Neither of these gestures, however, seems to be able to explain this asymmetry. The raise of the phrase curve is a global mechanism and has nothing to do with the asymmetry of the local interaction between intonation and the final tone. If the strength mechanism accounts for the asymmetry, at the sentence final position a high strength on Tone2 should be more difficult to identify than that on Tone4. Our perception test on focus identification, however, does not support this hypothesis. The mechanism of a focus must be a high strength, but the tone of the focused syllable, according to conclusion 8 above, does not affect focus identification under Statement, even if the focus is at the end of a sentence. The inability of the strength mechanism to explain this asymmetry implies either of the following two conclusions: 1. the strength mechanism is a 'false' one; 2. there exists another mechanism that accounts for the asymmetry. The first implication, however, is not tenable, as we can see from the discussion below as well as in section 3.

The third and fourth asymmetries are related to both focus and intonation type. The third asymmetry is that a focus at the end of a sentence makes statement intonation harder to identify but makes question intonation easier to identify. And from conclusion 6 and 7 we can generalize the fourth asymmetry: Question intonation makes a focus at the end of a sentence easier to identify whereas statement intonation does not. Both of the asymmetries are consistent with the strength mechanism of question intonation. Both question intonation and a final focus have a higher strength at the sentence final position. Therefore, presence of both in a sentence will make it easier for the listeners to identify the higher strength mechanism, which is an indicator of question intonation to the listeners in the intonation type identification test and an indicator of focus in the focus identification test. If there is a focus at the end of a statement, the higher strength of the last focused tone may be misinterpreted as a mechanism of question intonation for some listeners. Therefore more statements were identified as questions if focus was presented in final position.

In summary, the perception experiments found evidence for the strength mechanism of question intonation. It also found an asymmetry about the effects of the sentence-final tone2 and tone4 on question intonation identification. That the strength and the phrase curve mechanisms cannot explain this asymmetry suggests there is another mechanism of question intonation.

3 Evidence from Acoustic Analyses

Comparisons of statement and question intonation were made on the overall F_0 pattern, the overall intensity and duration pattern, the intensity and duration of the final tones, and the F_0 of the final Tone2 and Tone4. Only the unfocused utterances were used for the acoustic analyses.

3.1 Overall F_0 Pattern

Since each statement and question intonation pair has the same tone sequence, we can calculate the difference of the average F_0 over the tone at each syllable pair. The results are shown in Figure 2.

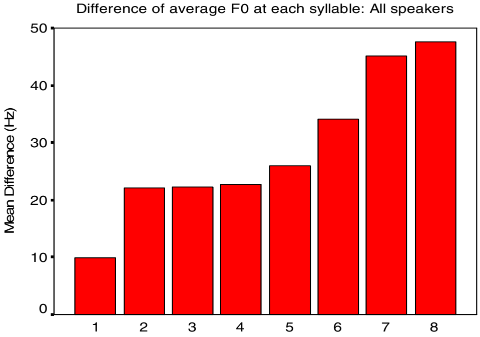


Fig. 2. Difference of the average F_0 over each syllable pair (question minus statement) for all speakers

Clearly, the F_0 curve of question intonation is higher overall than that of statement intonation. We can also see that the difference widens toward the end of the sentence.

This is consistent with the mechanisms of question intonation we found in our Stem-ML modeling study of Chinese intonation: Question intonation has a higher phrase curve and higher strengths at sentence final tones.

On the other hand, although in general the F_0 curve of question intonation is higher than that of statement intonation, Tone3 in a question may reach the same low point as in a statement at any sentence position, as we can see from Figure 1 above. To further support the conclusion that Tone3 in statement and question intonation may reach the same low point at any sentence position, Table 1 provides the counts of the occurrences where Tone3 reaches the same or almost the same low point (the difference is less than 5 Hz) at different sentence positions.

Table 1. Counts of the occurrences of the same low Tone3s in statements and questions

Speaker	Sentence position of Tone3			
	Syllable 1	Syllable 3	Syllable 5	Syllable 7
S1	2 (11)*	2 (16)	0 (20)	0 (14)
S2	2 (11)	1 (16)	0 (20)	0 (14)
S3	4 (11)	3 (16)	0 (20)	0 (14)
S4	3 (11)	1 (16)	2 (20)	1 (14)
S5	1 (11)	1 (16)	0 (20)	2 (14)
S6	9 (11)	11 (16)	6 (20)	5 (14)
S7	0 (11)	10 (16)	7 (20)	4 (14)
S8	6 (11)	10 (16)	7 (20)	2 (14)
Total	27 (88)	39 (128)	22 (160)	14 (112)

* The numbers in the brackets are the total occurrences.

3.2 Overall Duration and Intensity Pattern

Each utterance has eight syllables. Each syllable has its own duration and intensity. Duration is the time span of the syllable and intensity is measured by the highest intensity value of the syllable. The mean duration and intensity of each syllable in statement and question intonation across all speakers are shown in Figure 3 and 4 respectively.

Figure 3 shows that the syllables in question intonation are shorter than those in statement intonation in every position except the last syllable, which is longer in question intonation. It also shows final lengthening of both statement and question intonation, because of which the final syllable is longer than the other syllables.

Figure 4 shows that question intonation has a higher intensity curve than statement intonation and that the difference between them grows toward the end of the sentence.

The duration and intensity patterns shown in Figures 3 and 4 strongly support the strength mechanism: The sentence final tones have higher strength values in question intonation.

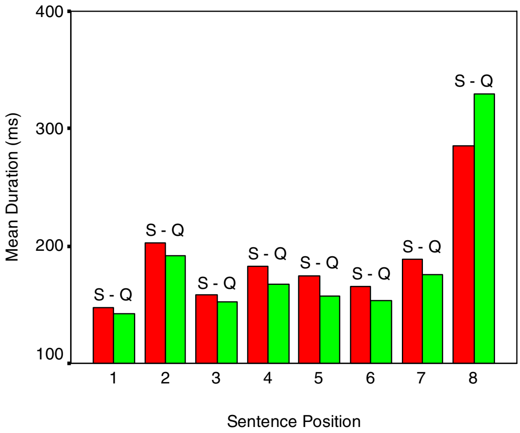


Fig. 3. Overall duration pattern of statement and question intonation (S: Statement; Q: Question)

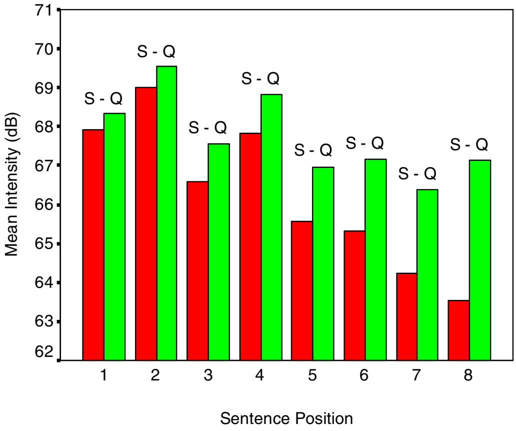


Fig. 4. Overall intensity pattern of statement and question intonation (S: Statement; Q: Question)

3.3 Duration and Intensity of the Final Tones

Figure 5 shows the duration and intensity difference between statement and question intonation for each of the sentence final tones.

From Figure 5 we can see that the final Tone3 and Tone4 are longer in question intonation than in statement intonation, whereas the final Tone1 and Tone2 have similar duration in the two intonation types.

In many cases the final Tone3 is only a low target in a statement. In a question, however, the rising end of Tone3 appears. Therefore, it is not surprising that the final Tone3 in question intonation is much longer than in the statement. However, why the final Tone4 is longer in question intonation whereas the final Tone1 and Tone2 are not is puzzling.

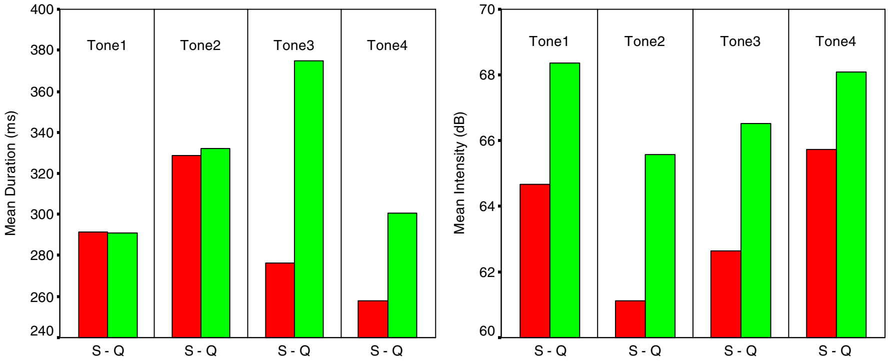


Fig. 5. Duration and intensity difference between statement and question intonation for each of the final tones (S: Statement; Q: Question)

We can also see from Figure 5 that each final tone has a higher intensity in question intonation than in statement intonation. The difference between them, however, is the largest for Tone2 and the smallest for Tone4.

3.4 F_0 of the Final Tone2 and Tone4

Both the perception experiments and the duration and intensity analyses above revealed an asymmetry between the final Tone2 and Tone4. Do they also show an asymmetry in F_0? This section tries to answer this question. Two F_0 parameters are extracted and compared for the final Tone2 and Tone4: F_0 at the end and F_0 slope. The results are shown in Figures 6 and 7.

From Figures 6 and 7 we can see that both the F_0 at the end and the F_0 slope of the final Tone2 are higher in question intonation than in statement intonation. The F_0 at the end of the final Tone4 is higher in question intonation than in statement intonation whereas the F_0 slope is not different between the two intonation types.

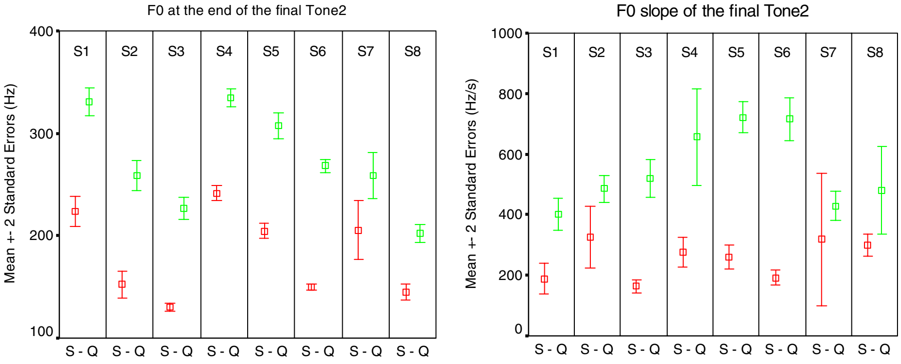


Fig. 6. F_0 of the final Tone2 (S: Statement; Q: Question)

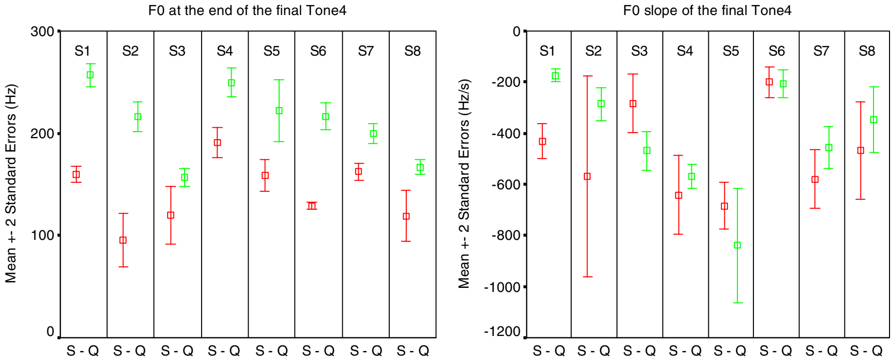


Fig. 7. F_0 of the final Tone4 (S: Statement; Q: Question)

3.5 Summary and Discussion

To summarize the results of the acoustic analyses, I draw the following conclusions: 1. The F_0 curve of question intonation is higher than that of statement intonation. Tone3, however, may sometimes pull the question curve down to the statement curve. 2. Question intonation has a higher intensity curve than statement intonation and the difference between them becomes greater toward the end of the utterance. 3. The final Tone3 and Tone4 are longer in question intonation than in statement intonation whereas the final Tone1 and Tone2 have similar duration in the two intonation types. 4. Each final tone has a higher intensity in question intonation than in statement intonation and the difference between them is the largest for Tone2 and the smallest for Tone4. 5. Both the F_0 at the end and the F_0 slope of the final Tone2 are higher in question intonation than in statement intonation. 6. F_0 at the end of the final Tone4 is higher in question intonation than in statement intonation whereas the F_0 slope is not different between the two intonation types.

The perception results in section 2 suggest that a third mechanism is needed to explain the asymmetry in the effects of the sentence-final Tone2 and Tone4 on question intonation identification. I will argue that the third mechanism is a tone-dependent mechanism functioning on the final tone.

The final tone in question intonation is strengthened by the strength mechanism, which expands the F_0 range of the final tone. If there is no other mechanism functioning on the final Tone4 in question intonation, the F_0 at its end should stay low, as under focus [14]. The acoustic analyses above, however, showed that the F_0 at the end of the final Tone4 is higher in question intonation than in statement intonation. Therefore, there must be another mechanism causing the F_0 at the end of the final Tone4 in question intonation to be raised, therefore higher than in statement intonation.

From the acoustic analyses we know that the F_0 at the end and the F_0 slope of the final Tone2 are greater in question intonation than in statement intonation. I propose that the third mechanism will increase the slope of the final Tone2 in question intonation if the slope can be naturally increased. Why can it not raise the F_0 at the end? According to previous studies [14], sentence final Tone2 has a longer duration under focus (strengthened). The sentence final Tone2 in question intonation (also strengthened), therefore, must be longer than in statement intonation if the third mechanism of question intonation does not affect the duration of the final Tone2. However, the acoustic analyses above show the contrary: The duration of the final Tone2 in question intonation is not longer than that found in statement intonation. If we assume that the third mechanism of question intonation raises the F_0 at the end of the final Tone2, it will be very difficult to explain why the final Tone2 in question intonation is not longer than in statement intonation. There is no such difficulty if we assume that changing the rising slope is the mechanism. If the slope is steeper, it will take less time to finish a *span*, which is, from the speakers' point of view, a pitch range that is large enough to indicate question intonation and also natural.

From the discussion above, we can see that the third mechanism of question intonation is different from the phrase curve mechanism and the strength mechanism in the following ways: First, it is a strictly local mechanism that functions only on the last tone; second, it is tone dependent; it flattens the falling slope of the final Tone4 and steepens the rising slope of the final Tone2. The intuition of the mechanism is simple: If there is a falling tone at the end of question intonation, the falling tone goes down more slowly; if there is a rising tone at the end of question intonation, the rising tone goes up more quickly. The idea that the realization of intonation type is sensitive to tonal identity was first developed in Shih (1988) [15]. Most of the Chinese intonation models in the literature do not, however, capture this aspect of tone and intonation interaction.

The tone-dependent mechanism may conflict with the strength mechanism on the final Tone2. This mechanism requires that the final Tone2 in question intonation go up more quickly or as soon as possible. The strength mechanism, however, requires that the beginning low part of the final Tone2 in question intonation be lengthened, or not go up soon [14]. This probably explains why question intonation is more difficult to realize if there is a Tone2 at the end of a sentence.

I have paid little attention to the F_0 contours of the final Tone1 and Tone3, partly because they are difficult to study: In an utterance, Tone1 can be either a target or a high-level contour and Tone3 can have either a rising end or not. Although the third mechanism is tone dependent, whether it functions on different versions of Tone1 (having a level contour or not) or Tone3 (having a rising end or not) in the same way or in different ways is neither clear nor a trivial question, and remains a question for further research.

4 Conclusions

Results from the perception experiments demonstrate that statement intonation is easier to identify than question intonation, and, while the tone of the last syllable does not affect statement intonation identification, it does affect question intonation identification. The intonation identification test was a forced choice test: listeners must identify the intonation type of each utterance as either statement or question intonation. That question intonation identification was less accurate means that many question intonation utterances were identified as statement intonation. This suggests that statement intonation is a default or unmarked intonation type; listeners fall back on this option when there is not enough information suggesting 'question.' The conclusion is also supported by the fact that the tone of the last syllable does not affect statement intonation identification. Question intonation, however, is a marked intonation type. It can only be identified if listeners actually hear the 'question' features/mechanisms.

Three mechanisms of question intonation have been proposed: an overall higher phrase curve, higher strengths of sentence final tones, and a tone-dependent mechanism that flattens the falling slope of the final falling tone and steepens the rising slope of the final rising tone. The phrase curve and strength mechanisms were revealed by the computational modeling study and verified by the acoustic analysis as well as the perceptual study: 1. Overall, the F_0 and intensity of question intonation are higher than statement intonation (phrase curve mechanism). 2. The F_0 and intensity difference between question intonation and statement intonation becomes higher toward the end of the sentence (strength mechanism). 3. The syllables in question intonation are shorter than those in statement intonation in every position except the last syllable, which is longer in question intonation (strength mechanism). 4. Focus at the end of a sentence makes statement intonation harder to identify but makes question intonation easier to identify. Question intonation makes focus at the end of a sentence easier to identify whereas statement intonation does not (strength mechanism). The third mechanism, a tone-dependent mechanism, was suggested by a result from the perceptual study: Question intonation is easier to identify if the sentence-final tone is falling whereas it is harder to identify if the sentence-final tone is rising. Neither the phrase curve mechanism nor the strength mechanism can explain this result. The phrase curve mechanism is global and tone-independent and the strength mechanism is partially global and also tone-independent. The asymmetry of the effects of the final Tone2 and Tone4, however, is local and tone-dependent. Acoustic analyses on the final Tone2 and Tone4 suggested that the tone-dependent mechanism of question intonation flattens the final falling tone and steepens the final rising tone.

The tone-dependent mechanism may conflict with the strength mechanism on the final Tone2: This mechanism requires that the final Tone2 in question intonation go up more quickly or as soon as possible. The strength mechanism, however, requires that the beginning low part of the final Tone2 in question intonation be lengthened, or not go up soon. This likely explains why question intonation is more difficult to realize as well as identify if there is a Tone2 at the end of a sentence.

Acknowledgments

I thank Abigail Cohn, Greg Kochanski, Amanda Miller-Ockhuizen, Mats Rooth, and Chilin Shih for their suggestions and comments on this project.

References

1. DeFrancis, J. F.: Beginning Chinese. New heaven: Yale University Press (1963)
2. Tsao, W.: Question in Chinese. Journal of Chinese Language Teachers' Association (1967) 15-26

3. Gårding, E.: Constancy and variation in Standard Chinese tonal patterns. Lund University Working Papers 28, linguistics-phonetics (1985) 19-51
4. Gårding, E.: Speech act and tonal pattern in Standard Chinese: constancy and variation. Phonetica (1987) 13-29
5. Shen, J.: Beijinghua shengdiao de yinyu he yudiao[Pitch range of tone and intonation in Beijing dialect]. in BeijingYuyin Shiyanlu, Lin, T.; Wang L. (ed.). Beijing: Beijing University Press (1985)
6. Shen, J.: Hanyu yudiao gouzao he yudiao leixing [Intonation structure and intonation types of Chinese]. Fangyan (1994) 221-228
7. Shen, X.: The Prosody of Mandarin Chinese, University of California Press (1989)
8. Peng, S., Chan, M., Tseng, C., Huang, T., Lee, O., Beckman, M.E.: Towards a Pan-Mandarin system for prosodic transcription. In: Sun-Ah Jun (ed.), Prosodic Typology: The Phonology of Intonation and Phrasing. Oxford University Press, Oxford, U.K. (2005) 230-270
9. Kochanski, G.P., Shih, C.: Prosody modeling with soft template. Speech Communication (2003) 311-352
10. Yuan, J., Shih, C., Kochanski, G.P.: Comparison of declarative and interrogative intonation in Chinese. In Proceedings of Speech Prosody 2002. Aix-en-Provence, France (2002) 711-714
11. Yuan, J., Shih, C.: Confusability of Chinese Intonation. In Proceedings of Speech Prosody 2004. Nara, Japan (2004) 131-134
12. Yuan, J.: Perception of Mandarin Intonation. Proceedings of ISCSLP 2004. Hong Kong (2004)
13. Liu F., Xu Y.: Parallel Encoding of Focus and Interrogative Meaning in Mandarin Intonation. Phonetica (2005) 70-87
14. Yuan, J.: Intonation in Mandarin Chinese: Acoustics,Perception, and Computational Modeling, Ph.D. Dissertation, Cornell University, Ithaca (2004)
15. Shih, C.: Tone and Intonation in Mandarin. Working Papers of the Cornell Phonetics Laboratory (1988) 83-109

Comparison of Perceived Prosodic Boundaries and Global Characteristics of Voice Fundamental Frequency Contours in Mandarin Speech

Wentao Gu[1,2], Keikichi Hirose[1], and Hiroya Fujisaki[1]

[1] The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
[2] The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong
{wtgu, hirose}@gavo.t.u-tokyo.ac.jp, fujisaki@alum.mit.edu

Abstract. Although there have been many studies on the prosodic structure of spoken Mandarin as well as many proposals for labeling the prosody of spoken Mandarin, the labeling of prosodic boundaries in all the existing annotation systems relies on auditory perception, and lacks a direct relation to the acoustic process of prosody generation. Besides, perception-based annotation cannot ensure a high degree of consistency and reliability. In the present study, we investigate the phrasing of spoken Mandarin from the production point of view, by using an acoustic model for generating F_0 contours. The relationship between perceived prosodic boundaries at various layers and phrase commands derived from the model-based analysis of F_0 contours is then revealed. The results indicate that a perception-based prosody labeling system cannot describe the prosodic structure as accurately as the model for F_0 contour generation.

Keywords: prosodic hierarchy, perceived prosodic boundary, F_0 contour, phrase, command-response model, Mandarin, perception, production.

1 Introduction

While the units and structures of a written message are largely well-defined within the framework of a given grammar, it is not the case for the units and structures of a spoken message, since they have traditionally been defined on the basis of subjective impressions, without referring to objective acoustic-phonetic characteristics of speech. In order to have definitions based on physically observable characteristics, Fujisaki defined prosody as the systematic organization of various linguistic units into an utterance, or a coherent group of utterances in the process of speech production [1]. As such, its units and structures can be described in terms of acoustic-phonetic characteristics of speech.

It turns out that the units and structures of prosody thus defined are not identical, though generally closely related to, the units and structures of syntax of the linguistic message that underlies the utterance. It is also known that the units of prosody and their acoustic-phonetic manifestations are not language-universal but somewhat language-specific. For instance, the units of prosody of an utterance of spoken Common Japanese were defined by Fujisaki et al. [2] to be prosodic word, prosodic phrase, prosodic clause, and prosodic sentence, on the basis of accent commands and

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 31 – 42, 2006.

phrase commands extracted from the voice fundamental frequency contour (i.e., F_0 contour), as well as pauses, but tone languages such as Mandarin certainly need different definitions. In the present paper, we shall look into the units and structures of prosody of spoken Mandarin.

Although there have been many studies on the prosodic structures of spoken Mandarin [3]-[8], no consensus has been attained on the hierarchy and the definitions of various prosodic units. This is in line with the fact that there is no consensus on the syntactic hierarchy of Chinese, especially considering the ambiguity in defining a 'word' as well as a certain degree of randomness in specifying the punctuations.

Among others, the C-ToBI system [7], [8] proposes a five-layer hierarchy: syllable (SYL), prosodic word (PW), minor prosodic phrase (MIP), major prosodic phrase (MAP), and intonation group (IG). In contrast, another ToBI-like system proposed by Tseng [3]-[5] defines a prosodic structure of six layers: syllable (SYL), prosodic word (PW), prosodic phrase (PPh) or utterance, breath group (BG), prosodic phrase group (PG), and discourse (DIS); the boundaries for these units, from small to large, are associated with the break indices B1 to B5, respectively. In these two systems, MAP and BG may approximately coincide, though they are defined from different perspectives.

However, the annotation of prosodic boundaries (break indices) in all these labeling systems relies more or less on auditory perception. This inevitably results in two problems.

First, the annotation based on perceived prosodic characteristics is irreversible. In other words, it is not possible to reproduce the prosodic characteristics of the original speech signals because annotation involves symbolization which discards quantitative information, nor is it possible to restore the underlying events in prosody production from the results of annotation because the inverse process cannot be operated. It is no doubt that strong relations exist between speech production and speech perception, and all the perceived prosodic characteristics should have the corresponding events in prosody production; however, there is by no means one-to-one correspondence, since both production (encoding) and perception (decoding) of prosody information are complex and multilayered (especially considering that perception is also affected by linguistic information). Besides, it is also concerned with the limited capability of human's perception of prosodic details (e.g., the local details of pitch movements [9]).

Second, the prosodic labels based on perception are subjective and the results may vary with transcribers. This is an intrinsic flaw of any ToBI-like systems [10], though one of their design goals was a high degree of inter-transcriber reliability. It was reported in [7] that the consistency between four transcribers in labeling break indices of C-ToBI on a read speech corpus is 78%, not to mention spontaneous speech. A similar test was also conducted for Tseng's labeling system [3], showing that inter-transcriber inconsistency is maintained even after the exchange of notes for labeling.

Thus, perception-based prosodic labeling systems cannot be used efficiently in the study of prosody generation for the purpose of speech synthesis.

In order to conduct a more objective investigation into the prosodic structure, we need to look into the acoustic evidences. There have been many studies on the acoustic cues for perceiving various prosodic boundaries. For example, it is shown in [11], [12] that pause, pre-boundary segment/syllable lengthening, and F_0 reset are major cues for prosodic boundaries, either in read speech or in spontaneous speech of Mandarin. Besides, they are correlated with or complementary to each other.

Among the three major cues, pause duration can be measured directly, and segment/syllable duration can also be reliably measured after an appropriate normalization to remove the segmental effects. The changes in F_0, however, have not been carefully investigated. Direct comparison of local F_0 values is not adequate even after a kind of tone normalization, because it is well known that the F_0 contour also contains the information of wide-range utterance intonation which directly reflects the organization of various prosodic units.

Based on these considerations, we shall investigate the phrasing of spoken Mandarin from the production point of view, by using a generative model for F_0 contours, which separates local lexical tones and global utterance intonation. On the basis of the same speech material, the relationship between the perceived prosodic boundaries and the model-based phrase commands can then be revealed.

2 The Model for the Generation of F_0 Contours

There are many approaches to analyzing F_0 contours, but only a few of them are capable of giving fully quantitative representations to F_0 contours. Since in the current study we are concerned with the perspective of production, a model for the process of generating F_0 contours should be the best choice. Therefore, we use the command-response model [1], [13] proposed by Fujisaki and his coworkers.

The command-response model for the process of F_0 contour generation describes F_0 contours in the logarithmic scale as the sum of phrase components, accent/tone components, and a baseline level $\ln F_b$. The phrase commands produce phrase components through the phrase control mechanism, giving the global shape of F_0 contours, while the accent/tone commands generate accent/tone components through the accent/tone control mechanism, characterizing the local F_0 changes. Both mechanisms are assumed to be critically-damped second-order linear systems. It has already been shown that phrase and accent/tone commands have good correspondence with various linguistic and paralinguistic information of speech.

The details of the model formulation are described in [1], [13]. Following the previous works, the constants α, β, and γ are set at their respective default values 3.0 (1/s), 20.0 (1/s), and 0.9 in the current study.

Unlike most non-tone languages that need only positive accent commands, tone languages usually require tone commands of both positive and negative polarities due to faster local tonal changes. For a specific tone language, a set of tone command patterns needs to be specified in the model.

Mandarin has four lexical tones: T1 (high tone), T2 (rising tone), T3 (low tone), and T4 (falling tone). Our previous work [14] has shown that the inherent tone command patterns for these four tones are: positive for T1, negative followed by positive for T2, negative for T3, and positive followed by negative for T4. Besides, Mandarin also has a so-called neutral tone (T0), which does not have an inherent tone command pattern; instead, the polarity of tone command for T0 varies largely with the preceding tone. Any lexical tones in Mandarin can be neutralized in an unstressed syllable.

If we disregard the minor effects of microprosody, an entire F_0 contour can be reproduced from a set of commands and parameters. Thus, unlike perception-based annotation, the modeling of F_0 contours is essentially a reversible process. Moreover, the underlying events in F_0 production can be induced in the framework of the model, because physically observable F_0 contours can be analyzed more closely with a mathematical process which inspects both local features and global trends.

3 Speech Data

The speech data used in the current study is a small portion of the COSPRO05 speech corpus of Taiwan Mandarin [16] collected and annotated by the Phonetics Lab, Institute of Linguistics, Academia Sinica, and kindly provided by Dr. Tseng. The speech corpus was recorded by two native speakers of Taiwan Mandarin (one male and one female) at their normal speech rates (4.33 and 4.23 syllables/sec, respectively). Both speakers were radio announcers under 35 years of age at the time of recording.

There are six discourses in the current speech data which are the reading of three paragraphs by the two speakers (each discourse here is a single paragraph). The three paragraphs consist of 93, 168, and 369 syllables, respectively.

The perception-based annotation of prosodic boundaries follows the hierarchy proposed in [3], purposely making no reference to lexical or syntactic properties. Five layers of boundaries, from small to large, are labeled with their respective break indices: B1, a syllable boundary at SYL layer where usually no break is perceived; B2, a perceived minor break at PW layer; B3, a perceived major break at PPh layer; B4, a perceived break at BG layer where the speaker is out of breath and about to take another full breath; and B5, a long break at PG layer after a perceived trailing-to-a-final-end. From B3 up, actual pauses occur at the boundaries.

The perceived prosodic boundaries were manually labeled by three trained transcribers independently. As stated in [4], intra- and inter-transcriber comparisons were conducted continually to ensure a high degree of intra- and inter-transcriber consistencies and hence the reliability of the labeled break indices. The annotation was not adopted until over 85% of inter-transcriber consistencies were attained.

The F_0 contours of the speech were extracted by the modified autocorrelation analysis of the LPC residual, and then were analyzed within the framework of the command-response model. During the model-based analysis, no reference was made to the prosodic annotation.

4 Model-Based Analysis

Although the inverse problem of the model, viz., induction of the underlying commands from a measured F_0 contour, is not analytically solvable, with the aid of linguistic information a linguistically meaningful solution can be derived by the method of analysis-by-synthesis, which includes two steps: manual initial estimation (automatic method is under development [17]), and successive approximation.

On the basis of the information of tone identity and syntactic structure, an initial estimation is conducted manually to deconvolve an F_0 contour into the underlying commands and a baseline frequency to give a solution in line with the linguistic constraints. For a constant style of read speech by a given speaker, the baseline frequency F_b can be initialized at a constant.

Basically, tone commands in each syllable should comply with the inherent command pattern of the specific tone type (various tone changes should be taken into consideration here, including tone *sandhi* and tone neutralization), though a pair of tone commands in T2 and T4 may be degraded into a single tone command in certain contexts of coarticulation in continuous speech.

The occurrences of phrase commands, on the other hand, are largely aligned with major syntactic boundaries and can be determined by comparison of local F_0 values between neighboring tones. Phrase commands are only assigned when necessary and

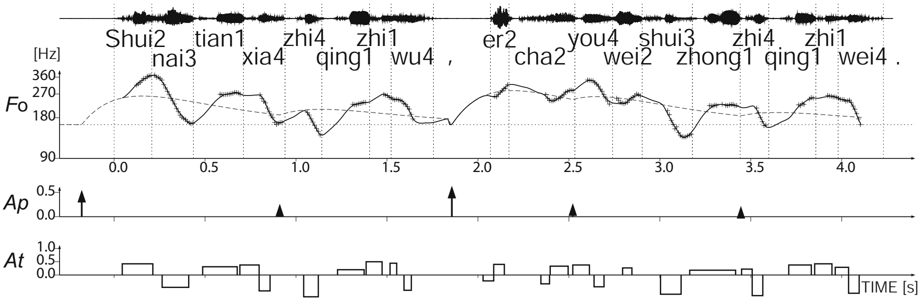


Fig. 1. Model-based analysis of the F_0 contour of the utterance: "水Shui2 乃nai3 天tian1 下xia4 至zhi4 清qing1 之zhi1 物wu4，而er2 茶cha2 又you4 为wei2 水shui3 中zhong1 至zhi4 清qing1 之zhi1 味wei4。" (Water is the clearest thing in the world, while tea has the purest taste in the water.)

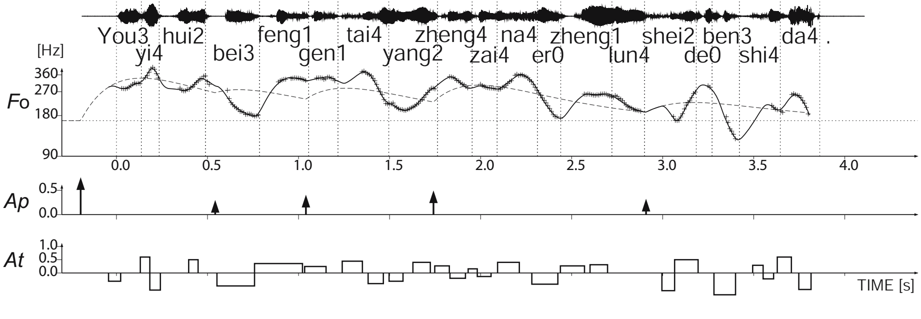


Fig. 2. Model-based analysis of the F_0 contour of the utterance: "有You3 一yi4 回hui2 北bei3 风feng1 跟gen1 太tai4 阳yang2 正zheng4 在zai4 那na4 儿er0 争zheng1 论lun4 谁shei2 的de0 本ben3 事shi4 大da4。" (Once the north wind and the sun were there arguing who was more capable.)

linguistically meaningful. For example, at many positions, whether to add a very small phrase command or not usually has little effect on the accuracy of approximation; in this case, we do not add it.

The details of the procedure can be referred to in [15], which is on the modeling of Cantonese but the principle is the same. After initial estimation, successive approximation is conducted through a hill-climbing search in the space of model parameters to obtain an optimal solution giving the least RMS error between the measured and the approximated F_0 contours in the logarithmic domain.

Even if initial estimation is done manually, a better inter-analyst consistency can be attained than for perception-based labeling systems, because the analysis is based on objective measurements instead of subjective impressions. Moreover, successive approximation decreases the inconsistency in initial estimation automatically.

Since each discourse in the current study was read in a constant style, the variation of the value of F_b in a discourse is constrained within a range of 10 Hz. For a pause longer than 0.3 sec, an F_0 reset is assigned at the beginning of the pause.

Figures 1 and 2 show the results of analysis on the F_0 contours of two utterances read by the female speaker, respectively. The crossed symbols indicate the measured F_0 values, while the solid lines, the dotted lines, and the dashed lines indicate the approximated F_0 contours, the baseline frequencies, and the contributions of phrase components, respectively. The difference between the approximated F_0 contour and the phrase components corresponds to the tone components.

The utterance in Fig. 1 has a pause of 0.3 sec between the two clauses, and hence F_0 is reset to the baseline value at the end of the first clause; for both clauses, the clause-initial phrase commands are conspicuously larger than the others. In Fig. 2, the utterance does not have any internal pauses, and the utterance-initial phrase command is notably larger than the others. In both figures, all the utterance-medial phrase commands coincide with certain syntactic boundaries, but they do not follow the hierarchy of syntax. For the two utterances, the RMS errors between the measured and the approximated $\ln F_0$ values within the voiced intervals are 0.015 and 0.012 respectively, equivalent to a relative error of 1.5% and 1.2% in F_0, indicating a very high accuracy of approximations of F_0 contours attained by the model.

5 Analysis Results

In most cases, the time for occurrence of a phrase command corresponds with (to be exact, slightly before) the segmental onset of a prosodic boundary at a certain layer; but not vice versa. Among the total of 282 phrase commands in the data, there are only 12 exceptions which do not occur at any labeled boundary – in 8 instances out of them, the phrase command is only one syllable apart from a boundary. These 8 exceptional instances are listed below, where the slashes indicate the labeled PW boundaries (B2),[1] while the vertical lines indicate the approximate positions of phrase commands we have detected.

[1] In fact, there still lacks a very clear definition of 'prosodic word' in the multi-layer labeling system, as in most other works on Mandarin. It also increases the ambiguity in labeling.

The male speaker:
｜原先 / 他是因 / 嫉妒 / 而｜激动
｜太阳｜就 / 出来｜/ 使劲 / 一晒
The female speaker:
｜原先 / 他是｜因 / 嫉妒 / 而激动
｜而 / 茶｜又为 / 水中｜/ 至清 / 之味
｜有一回｜/ 北风｜跟 / 太阳｜/ 正在 / 那儿 / 争论｜/ 谁的 / 本事大
｜他们俩 / 就｜商量 / 好了
｜只好 / 就｜算了
｜所以 / 北风｜/ 不得 / 不｜承认

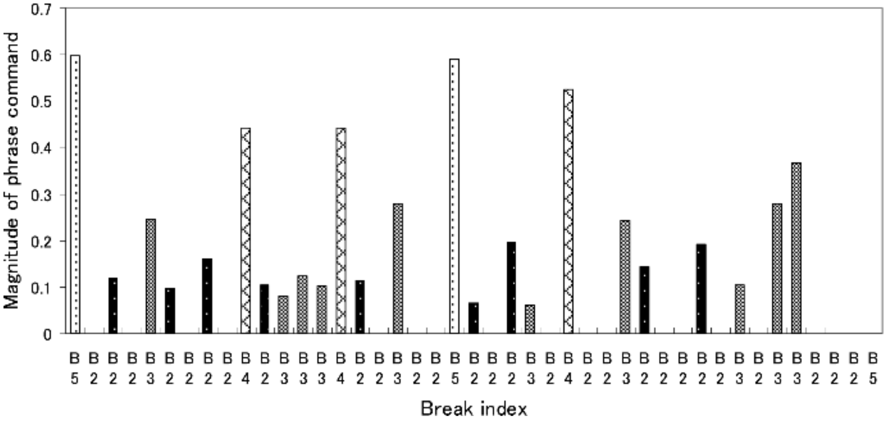


Fig. 3. Magnitudes of phrase commands at the prosodic boundaries in Paragraph 1 read by the male speaker

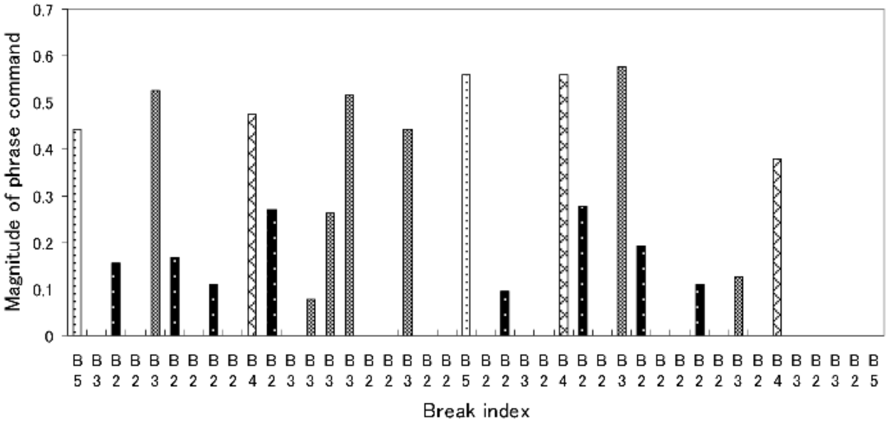


Fig. 4. Magnitudes of phrase commands at the prosodic boundaries in Paragraph 1 read by the female speaker

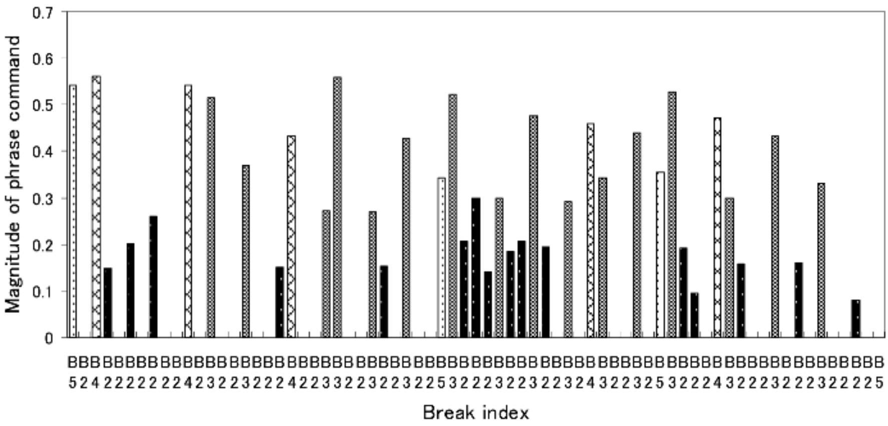


Fig. 5. Magnitudes of phrase commands at the prosodic boundaries in Paragraph 2 read by the male speaker

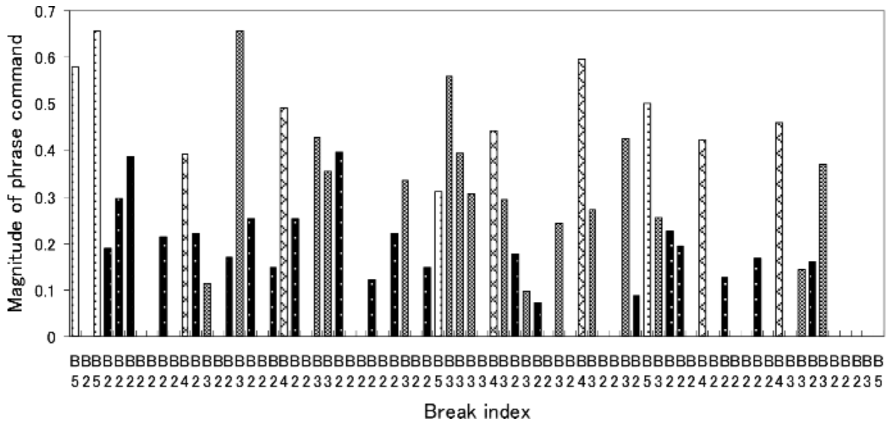


Fig. 6. Magnitudes of phrase commands at the prosodic boundaries in Paragraph 2 read by the female speaker

In each of these instances, there is a phrase command which is about one syllable apart from the nearest PW boundary – this syllable between the command and the boundary is always a monosyllabic syntactic word, and in most cases it is a function word like 而, 因, 跟, 就, 不. In fact, there is always an ambiguity in grouping these monosyllabic syntactic words into prosodic units, especially when judged by perception.

From the data including the above instances, we also observe that many PW boundaries are not accompanied by a phrase command. Hence, the claim that F_0 reset is a major cue for prosodic word boundary [7] is not supported; in fact, it also contradicts with the results on the comparison of mean F_0 values in the pre- and post-boundary syllables as given in [12]. We conjecture that some other cues may play more important role in perceiving PW boundaries. Meanwhile, we believe that the assignment of PW boundaries is more or less affected by syntactic information.

Figures 3 to 6 show the magnitudes of phrase commands at various layers of prosodic boundaries in four discourses, respectively. Here the phrase commands that are only one syllable apart from the corresponding boundaries are also counted. It is observed that the prosodic structures of the speech for the same text are different between the two speakers, though a certain degree of similarity exists.

Although there is an overall tendency that higher-layer boundaries correspond to larger phrase commands, considerable overlaps in the magnitude of phrase command are observed between the boundaries of different layers, especially between B3, B4, and B5.

To give a better view, Tables 1 and 2 show the statistics of boundary breaks at various layers as well as of the corresponding phrase commands, for the two speakers, respectively. Both the mean and the standard deviation of the magnitude of phrase command are given. The values given in the parentheses indicate the statistics calculated only on non-zero phrase commands.

Although higher-layer prosodic boundaries B4 and B5 are always accompanied by a phrase command, it is not always the case for lower-layer prosodic boundaries, especially for PW boundary (B2) which only shows a rate of occurrence of phrase commands at 36% and 29% for the two speakers respectively.

For prosodic phrase boundaries (B3), most of them (93% for the male speaker and 86% for the female speaker) are accompanied by a phrase command. Moreover, by looking closely into the 15 instances that are not accompanied by a phrase command,

Table 1. Statistics of prosodic boundaries and the corresponding phrase commands in the data of the male speaker

Break Index	Num. of labeled breaks	Num. of phrase commands	Mean of magnitude	Std. of magnitude
B2	170	61	0.06 (0.16)	0.08 (0.06)
B3	69	64	0.31 (0.33)	0.17 (0.15)
B4	15	15	0.46	0.12
B5	8	8	0.52 *	0.11 *

* If counting the phrase command immediately after the filler in the case where the PG begins with a filler, the mean and the standard deviation at B5 are 0.58 and 0.05, respectively.

Table 2. Statistics of prosodic boundaries and the corresponding phrase commands in the data of the female speaker

Break Index	Num. of labeled breaks	Num. of phrase commands	Mean of amplitudes	Std. of amplitudes
B2	177	51	0.05 (0.18)	0.09 (0.08)
B3	69	59	0.31 (0.36)	0.19 (0.16)
B4	15	15	0.48	0.08
B5	9	9	0.49 *	0.10 *

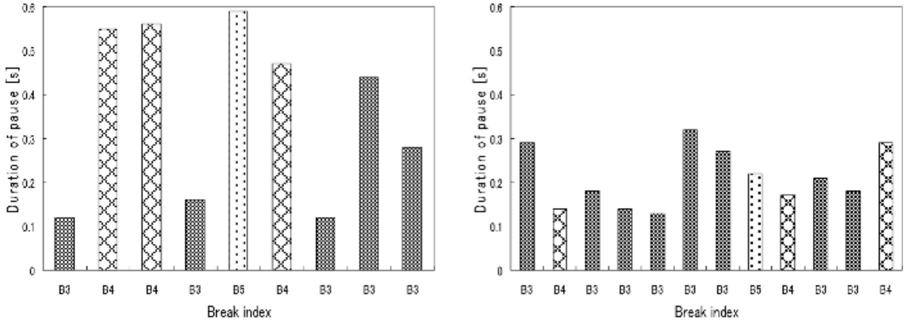
* If counting the phrase command immediately after the filler in the case where the PG begins with a filler, the mean and the standard deviation at B5 are 0.51 and 0.09, respectively.

we found that at least 10 of them should be labeled as B2 instead of B3, according to the definition that B3 be accompanied by a pause [3]-[5]. The mislabeling results from the confusion between physical ‘pause’ and perceived ‘break’ – the latter may result from other cues such as duration lengthening, intensity change, or F_0 reset. Especially, in many of these instances the closure of a stop was regarded mistakenly as a pause; in fact, a similar problem in labeling C-ToBI has also been noted in [12].

From these observations, we recognize that the perception-based labeling of prosodic boundaries is not always reliable, even if a few trained transcribers have attained a high degree of consistency. Meanwhile, the confusion between B2 and B3 indicates that there still lacks an explicit definition of ‘prosodic word’ and ‘prosodic phrase’ for spoken Mandarin, as also noted in the study of C-ToBI [7]. In our view, prosodic phrase boundaries are always accompanied by phrase commands, though the occurrence of a phrase command does not necessarily indicate a prosodic phrase boundary – it can also occur at certain prosodic word boundaries. In this sense, the model-based analysis can also be used to verify the annotation of break indices.

On the whole, phrase commands tend to be larger at higher-layer boundaries, though the ranges of magnitudes overlap considerably. A notable exception is that no significant difference is observed between B4 and B5. Nevertheless, when we looked more closely into the data, we found that the results for B5 could be corrected slightly. Two PGs in the data of the male speaker and one PG in the data of the female speaker begin with fillers like “OK” or “soon.” Since fillers usually have very low F_0 values, in the study of phrasing we can skip them and count the magnitude of the phrase command immediately after the filler instead. In this way, the average phrase command at B5 becomes slightly larger, as indicated below the two tables. Hence, for the male speaker it is larger than at B4, but for the female speaker the difference between B4 and B5 is still negligible.

The big overlaps between the magnitudes of phrase commands at B3, B4, and B5 indicate that F_0 is not the major cue for distinguishing between these layers. Then the question is: what are the major cues for identifying these break indices? Here, we also investigate pause duration at these perceived boundaries, because pause is well recognized to be an important cue for prosodic structure of spoken Mandarin, and it is also suggested in [3]-[5] that B5 is associated with a perceived longer break than B4.



(a) The male speaker.

(b) The female speaker.

Fig. 7. Duration of pauses at the prosodic boundaries in the discourses of Paragraph 1

Figure 7 shows the duration of pauses at B3, B4, and B5 boundaries in the discourses of Paragraph 1 by the two speakers. Among B3 labels in the data, only those accompanied by an actual pause are counted here. Although panel (a) for the male speaker shows a tendency that higher-layer boundaries are associated with longer pauses, panel (b) for the female speaker shows a big overlap in the distribution of pause duration at different layers of boundaries. Hence, pause duration is not always an important cue for identifying the break indices. We conjecture that several acoustic cues work jointly in determining the layers of perceived prosodic boundaries, and the information of syntax and semantics may have an impact as well. In fact, there still lacks an explicit and objective criterion for identifying different layers of prosodic units in spoken Mandarin.

6 Conclusions

On the basis of the command-response model for the process of F_0 contour generation, we have analyzed the distributions of phrase commands at various layers of perceived prosodic boundaries in continuous speech of Mandarin.

The prosodic labeling system proposed by Tseng [3]-[5] was used as a test case, but the results can be applied similarly to any other perception-based labeling systems like C-ToBI [7], [8] since they differ only slightly in the definitions of prosodic units. Likewise, we used the command-response model as the best choice for studying the production of F_0 contours, but we believe that similar conclusions will be reached with any other production-based (i.e., objectively-measured) approaches.

Our analysis has shown that there is only partial and qualitative correspondence between the production of phrase commands and the perception of prosodic boundaries. On the one hand, the majority of phrase commands correspond with a prosodic boundary at a certain layer. It indicates that most phrase commands give rise to a perceived prosodic boundary. On the other hand, only higher-layer prosodic boundaries are always accompanied by a phrase command, while it is not always the case for lower-layer prosodic boundaries, especially for prosodic word boundary (B2). The perception of prosodic word boundaries may be attributed more to the cues other than pitch, and may also be affected more or less by syntactic information.

Likewise, the magnitude of phrase command is only partially correlated with the layer of perceived boundary. Although an overall tendency is observed that higher-layer boundaries tend to be accompanied by larger phrase commands, the ranges of phrase command magnitudes overlap considerably between different layers of boundaries, especially between B3, B4, and B5.

In principle, prosodic boundaries are perceived on the basis of an integration of various acoustic cues such as pitch, duration, intensity, and pause, which are sometimes mutually complementary. Hence, perceived boundaries cannot give an explicit description of each individual prosodic feature, which however is required for synthesizing speech signals. To make the things worse, the annotation of perceived prosodic boundaries lacks a high degree of reliability due to its subjective nature.

Although the perception-based annotation of prosodic boundaries is helpful for the study of prosody perception, it cannot be used directly in the study of prosody generation for the purpose of speech synthesis, which requires a fully quantitative description of the physically observable prosodic characteristics. In order to have a better definition as well as a deeper understanding of the hierarchical structure for the prosody of spoken Mandarin, we should study the issue not only from the perception perspective (subjectively) but also from the production perspective (objectively).

Acknowledgements. We wish to thank Dr. Chiu-Yu Tseng at the Phonetics Lab, Institute of Linguistics, Academia Sinica, for discussion and kindly providing the annotated speech data.

References

1. Fujisaki, H.: Prosody, Models, and Spontaneous Speech. In: Sagisaka, Y., Campbell, N., Higuchi, N. (eds.): *Computing Prosody*. Springer-Verlag, New York (1997) 27-42.
2. Fujisaki, H., Kawai, H.: Realization of Linguistic Information in the Voice Fundamental Frequency Contour of the Spoken Japanese. *Proc. ICASSP 1988*, New York, NY (1988) vol. 2: 663-666.
3. Tseng, C., Chou, F.: A Prosodic Labeling System for Mandarin Speech Database. *Proc. 14th ICPhS*, San Fransisco, CA (1999) 2379-2382.
4. Tseng, C., et al.: Fluent Speech Prosody: Framework and Modeling. *Speech Communication* 46 (2005) 284-309.
5. Tseng, C.: Higher Level Organization and Discourse Prosody. *Proc. TAL 2006*, La Rochelle, France (2006) 23-34.
6. Cao, J.: Rhythm of Spoken Chinese – Linguistic and Paralinguistic Evidences. *Proc. ICSLP 2000*, Beijing, China (2000) vol. 2: 357-360.
7. Li, A., et al.: Speech Corpus of Chinese Discourse and the Phonetic Research. *Proc. ICSLP 2000*, Beijing, China (2000) vol. 4: 13-18.
8. Li, A.: Chinese Prosody and Prosodic Labeling of Spontaneous Speech. *Proc. Speech Prosody 2002*, Aix-en-Provence, France (2002) 39-46.
9. Clark, J., Yallop, C.: An Introduction to Phonetics and Phonology. 2nd edn. Blackwell Publishers, Cambridge, MA (1995).
10. Syrdal, A., McGorg, J.: Inter-Transcriber Reliability of ToBI Prosodic Labeling. *Proc. ICSLP 2000*, Beijing, China (2000) vol. 3: 235-238.
11. Li, A.: An Acoustic Analysis on Prosodic Phrases and Sentence Accents in Mandarin Dialogues. *Proc. 4th National Conference on Modern Phonetics*, Beijing, China (1999).
12. Liu, Y., Li, A.: Cues of Prosodic Boundaries in Chinese Spontaneous Speech. *Proc. 15th ICPhS*, Barcelona, Spain (2003) 1269-1272.
13. Fujisaki, H.: Information, Prosody, and Modeling – with Emphasis on Tonal Features of Speech. *Proc. Speech Prosody 2004*, Nara, Japan (2004) 1-10.
14. Fujisaki, H., Wang, C., Ohno, S., Gu, W.: Analysis and Synthesis of Fundamental Frequency Contours of Standard Chinese Using the Command-Response Model. *Speech Communication* 47 (2005) 59-70.
15. Gu, W., Hirose, K., Fujisaki, H.: Modeling the Effects of Emphasis and Question on Fundamental Frequency Contours of Cantonese Utterances. *IEEE Trans. Audio, Speech and Language Processing* 14 (2006) 1155-1170.
16. http://www.myet.com/COSPRO
17. Gu, W., Hirose, K., Fujisaki, H.: A General Approach for Automatic Extraction of Tone Commands in the Command-Response Model for Tone Languages. *Proc. Speech Prosody 2006*, Dresden, Germany (2006) 153-156.

Linguistic Markings of Units in Spontaneous Mandarin

Shu-Chuan Tseng

Institute of Linguistics, Academia Sinica, Nankang 115, Taipei
tsengsc@gate.sinica.edu.tw

Abstract. Spontaneous speech is produced and probably also perceived in some kinds of units. This paper applies the perceptually defined intonation units to segment spontaneous Mandarin data. The main aim is to examine spontaneous data to see if linguistic cues which mark the unit boundaries exist. If the production of spontaneous speech is a kind of concatenation of these "chunks", we can deepen our understanding of human language processing and the related knowledge about the boundary markings can be applied to improve language models used for automatic speech recognizers. Our results clearly show that discourse items and repair resumptions, which are typical phenomena in spontaneous speech, are mostly located at the boundary of intonation unit. Moreover, temporal marking of items at unit boundary is empirically identified through a series of analyses making use of segmentation of intonation units and measurements of syllable durations.

Keywords: Spontaneous speech, repairs, discourse items, tempo variability.

1 Introduction

To understand spontaneous speech production means more than to recognize the phonetic details of the speech signal. The final aim is to understand the content, i.e. the meaning of the signal. For human communication, a variety of cues are used to help the understanding of speech, e.g. world knowledge, prosody, gesture, and mimic. But for automatic speech recognizer, the only source comes from the speech signal itself. But spontaneous speech is continuous and more complicatedly it contains a lot of reductions and merging. Especially, interruptions and incomplete utterances often occur. Therefore, to deal with understanding of spontaneous speech production cannot only be associated with detection of individual phonetic information, but rather with information retrieval from the highly reduced linguistic information. The task is how to obtain the most useful cues from the limited phonetic content. For extracting units of meaning, an intermediate unit between word and utterance for segmenting spontaneous speech is necessary, because a unit with optimal length which at the same time deals with phonetic and semantic contents is required. This paper adopts the concept of intonation units from the field of conversation analysis, which are defined according to perceptual judgment of prosodic and semantic phrasing. Our aim is to test whether the unit boundary is marked by lexical and temporal cues specifically. In conversation, discourse items consisting of particles, markers, and fillers, normally mark locations within utterances where discourse functions need to be revealed. Repairs are important phenomenon in spontaneous speech and location

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 43 – 54, 2006.

of re-initiation of repairs also directly indicates a processing unit boundary. Moreover, changes in tempo provide clear cues to prosodic phrasing and furthermore indicate the presence of unit boundaries. In this section, literature review in the field of intonation units and tempo variability as well as discourse items in spoken Mandarin will be summarized.

1.1 Intonation Units

Various terms have been used for unit segmentation in the spoken language, for instance intonation phrases, intonation units, and turn constructional unit. They all base on a same pre-assumption that the prosodic structure in the spoken language does not necessarily correspond to the grammatical structure proposed for the written language. Intonation units are selected to segment our data, because they are associated with semantic, intonation and pragmatic information in conversation [7], [18], and our data have the form of conversation. Hirst and Bouzon [10] use z-scores of segment duration to study the lengthening effect of segments in stressed and unstressed words, in different positions within words and intonation units. In their result of British English, the effect of final lengthening of segment duration on the intonation unit boundary is clearly found, but not in word boundary. This empirically approves the statement that the final item of an intonation unit is often lengthened. Similarly, Dankovicova [5] analyzes the rhythmic patterns of spoken Czech. The duration of phonological words is measured and it was found that the first phonological word within an intonation phrase is usually spoken faster and final lengthening in terms of phonological words is also found at intonation phrase-final positions. These studies provide empirical supports for the relevance of the domain "intonation units" in the analysis of tempo variability. More importantly, in the study of prolongation in Mandarin conversation carried out by Lee *et al.* [11], final lengthening occurs more at the constituency units "word" and "phrase". At the level of "sentence", lengthening is produced mostly in sentence-medial position. Also, in modern Mandarin, the majority of words are mono- or disyllabic, so it is not especially surprising that prolongation is often located at the word boundary. Only the intermediate unit equivalent to "phrase" is left as a possible candidate. Based on the properties of intonation units mentioned above, intonation units are chosen to segment the spontaneous Mandarin data for later analyses.

1.2 Tempo Variability

Empirical evidence have been obtained, indicating the Czech has a more or less rallentando timing pattern (slowing down), where English tends to have an accelerando pattern [4]. Final lengthening is significantly more often observed in Singapore English than in British English [14]. These results imply that different languages, or even the same language, but spoken in different speaker communities, may have different timing templates governing the speech production. Furthermore, pause structure and boundary features are also important in communicative interaction, and may be determined spontaneously along the course of conversation [21]. In conversation, restarts, including repairs and repetitions, occur very often. Also relevant to spontaneous speech production, a reduced articulation rate in the case of disfluency has been reported in [16]. Fowler and Housum [8] mention that repeated words should require less processing time and therefore are uttered faster. Tseng [19] states in a study on the same set of data used in this paper that the re-initiation of repairs in Chinese is marked by a fast speaking tempo and a weaker intensity in comparison to the comparable position in the reparandum. The fact that the f0 values at the beginning of the reparandum and the alteration are not significantly different indirectly indicates a pitch reset. The features of the re-initiation process of speech repairs are similar to those of a new intonation unit relative to the preceding one. Therefore, in one of the subsequent analyses we will investigate the interaction of the repair resumption point and the intonation unit boundary. Also, temporal markings within repeated words are presented in many studies [3], [17], [12]. One reason may be that when there is a need to strengthen the difference between the new and given information, durational differences provide an efficient help for a quick and emphasized contrast [8]. More directly related to spontaneous speech production, O' Shaughnessy [15] finds that three factors are regarded relevant to word durations in spontaneous speech: the number of phonemes in each word, whether it is a function word or a content word, and whether the word forms part of a common sequence of words. Regarding the relation of duration to syntactic units, he found a tendency for words at the beginning of a syntactic unit to be slightly shorter than those at the end of the unit.

1.3 Discourse Items in Mandarin

Discourse items form a considerable part in Mandarin conversation. In Lee *et al.* [11], discourse particles and markers are the word class on which the prolongation is most frequently produced. Chao [1] states that grammatical and discourse types of particles are produced in neutral tones and interjections are more associated with intonational patterns. In Li and Thompson [13], six particles are discussed in detail. But no clear differentiation is made between particles serving grammatical functions and particles serving mainly pragmatic function. For processing our data, discourse particles in Mandarin are regarded as items whose pragmatic functions in discourse play an essential role other than syntactic structures or lexical meanings. They rarely contain concrete, substantial, lexical meaning. More importantly, in the Chinese writing system, some of the discourse particles are written in conventionalized characters which also appear as lexical entries in lexicon. Grammatical particles are normally unstressed, whereas discourse particles are often accompanied with an emphasized pitch type, prominent intensity or duration realization to express the pragmatic meaning they are supposed to deliver to the recipients. While processing our spontaneous Mandarin data, we are encountered with a wide variety of discourse items used in Taiwan. Because the definition and the difference of particles is elusive in the literature [1], [13], we propose here a system of discourse items consisting of discourse particles, discourse markers, and fillers, but excluding particles serving grammatical functions such as aspect marker (*le*), question marker (*ma*), structure particle (*de*) etc.

Table 1. This list contains discourse items in spoken Taiwan Mandarin. Please note that **dp** stands for discourse particles, **dm** for discourse markers, and **fl** for fillers. With regard to the column origin, **M** and **S** stand for Mandarin and Southern-Min, respectively.

Item	Written form	Group	Origin	Item	Written form	Group	Origin	Item	Written form	Group	Origin
A	啊	**dp**	**M**	*MA*	嘛	**dp**	**M**	*NEI*	-	**dp**	**M**
AI YA	哎呀	**dp**	**M**	*NOU*	喏	**dp**	**M**	*ON*	-	**dp**	**M**
AIYOU	唉呦	**dp**	**M**	*O*	喔哦	**dp**	**M**	*EIN*	-	**dp**	**S**
BA	吧	**dp**	**M**	*OU*	噢	**dp**	**M**	*HAN*	-	**dp**	**S**
E	呃	**dp**	**M**	*WA*	哇	**dp**	**M**	*HEIN*	-	**dp**	**S**
EN	嗯	**dp**	**M**	*WA SAI*	哇塞	**dp**	**M**	*HO*	-	**dp**	**S**
HAI	嗨	**dp**	**M**	*YE*	耶	**dp**	**M**	*UHN*	-	**fl**	**-**
HE	呵	**dp**	**M**	*YI*	咦	**dp**	**M**	*UHM*	-	**fl**	**-**
HEI	嘿	**dp**	**M**	*YOU*	呦	**dp**	**M**	*NHN*	-	**fl**	**-**
HWA	嘩	**dp**	**M**	*AI YE*	-	**dp**	**M**	*MHM*	-	**fl**	**-**
LA	啦	**dp**	**M**	*EI*	-	**dp**	**M**	*NAGE*	那個	**dm**	**M**
LIE	咧	**dp**	**M**	*HEN*	-	**dp**	**M**	*NA*	那	**dm**	**M**
LO	囉	**dp**	**M**	*HON*	-	**dp**	**M**				

Table 1 lists important discourse items in Taiwan Mandarin with information about the transcription, the corresponding character, the classification, and the origin. All these discourse items are not associated with any fixed lexical tones. Instead, the pitch types are directly associated with particular, pragmatic functions. Filled pauses such as *uhn* in English have counterparts in Mandarin, i.e. the particle *EN* and the filler *UHN*. *EN* has the main function of hesitation and is conventionally written as 嗯, whereas *UHN* is often used in conversation for signaling response. Discourse markers include word or word sequences whose original, syntactic function and content are lost in conversation, but used more for discourse purposes, e.g. hesitation. To take the word sequence determiner + classifier (*NA+GE*) as example, *NAGE* (*well*) is often used in conversation to make the utterance sound continuous, but without any syntactic role. When used for this purpose, it sometimes precedes a proper noun, which is redundant and ill-formed for determiner and classifier.

2 Data and Methodology

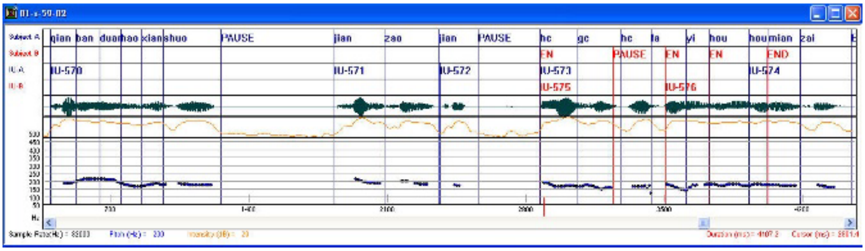
The data used for this study are extracted from the Taiwanese Putonghua Corpus (TWPTH) issued by the Linguistic Data Consortium [6]. The language under investigation is Taiwan Mandarin. Three free conversations are annotated in terms of intonation units and repairs. Detailed information about the speakers can be found in [20]. The definition of intonation units is based on the principles given in [18]. An intonation unit is regarded as a sequence of words which provide a coherent meaning for the annotators perceptually. Similar definitions of phrasing are often applied in speech synthesis to make a TTS system sound more naturally [9]. The first segmentation of intonation units has been done by the author. A second annotation by another annotator achieved about 85% consistency with the first annotation. The final data used for this study is the result of intensive discussions between the two annotators. The annotation of speech repairs follows those principles given in [19]. In order to process the data more efficiently, we have merged immediately adjacent, simple repairs into complex repairs. So the total number of the repairs is slightly different from that given in [19]. **Table 2** summarizes the general statistics of the data analyzed in this paper.

Table 2. Data summary

Subjects	**1-A**	**1-B**	**2-C**	**2-D**	**3-E**	**3-F**
# of syllables	2,713	586	1,381	1,448	1,787	803
# of identified repairs	96	24	39	50	52	28
# of identified units	578	181	314	296	345	148
# of monosyllabic units (%)	91(15.7%)	43(23.8%)	48(15.3%)	61(20.6%)	24(7%)	15(10.1%)
# of polysyllabic units (%)	487(84.3%)	138(76.2%)	266(84.7%)	235(79.4%)	321(93%)	133(89.9%)
# of units with final pauses(%)	198(34.3%)	37(20.4%)	158(50.3%)	141(47.6%)	210(60.9%)	80(54.1%)

Meaningful speech stretches containing repairs are manually cut out from the sound files of the dialogues and annotated by using the software PitchWorks developed by SCICON R & D. The segmented sound files are then manually labeled in two tiers: syllables and intonation units. The labeling of the sound data was checked by a second labeler. One important thing to note is that the annotation works of intonation units and repairs are done in two different stages independently. Subsequently, the two kinds of annotations were merged into a syllable-based database consisting of acoustical measurement results and boundary position marks of repair and intonation units. **Fig. 1** demonstrates the data annotation of the following example.

1-A：(前半段好像說PAUSE)$_{IU}$ (建造)$_{IU}$ (建PAUSE)$_{IU}$ (合格
(Qian2ban4duan4 hao3xiang4shuo1 PAUSE) (jian4zao4) (jian4 PAUSE) (he2ge2
(the first phase it seems PAUSE) (construct) (cons-PAUSE) (qualified
1-B：(EN)$_{IU}$
1-A：合法以後)$_{IU}$ (後面再)$_{IU}$
he2fa3 yi3hou4) (hou4mian4 zai4)
legal afterwards) (after that then)
1-B：(EN EN)$_{IU}$



Sample Rate(Hz) = 8000 Pitch (Hz) = 200 Intensity (dB) = 20 Duration (ms) = 4107.2 Cursor (ms) = 3881.4

Fig. 1. Boundaries of syllables and units of two speakers are annotated in different tiers

In the example, there are seven intonation units and two speech repairs. Speech repairs are not annotated to the sound files directly, but integrated into the merged syllable-based database mentioned above. "Jian4zao4 jian4" and "he2ge2 he2fa3" are annotated as two speech repairs. As a result, the former one is produced as two intonation units, where the latter one is produced in one single intonation unit. Making use of the labeled data, we investigate lexical and temporal cues at the boundary. Lexical cues include discourse items and repair resumption items. Temporal cues refer mainly to the items identified at the boundary. If linguistic evidence can be found marking the boundaries, it implies that intonation unit is a suitable unit which may reflect the process of language planning and production.

3 Results

3.1 Lexical Cues - Discourse Items

Discourse items are always associated with certain discourse functions. Assuming that intonation units can be regarded as a production (processing) unit for spontaneous speech, the type and location of discourse items in relation to their position in intonation units may reflect characteristics of their discourse functions. **Table 3** lists types of discourse items found in the data. In a study on discourse particles of Mandarin done on the Mandarin Conversational Dialogue Corpus (MCDC), it was found that discourse items are mainly located in utterance-medial positions in spontaneous speech, when the investigation domain is set to be "utterance", instead of "intonation unit" [20]. Although it is often argued that discourse particles are utterance-final for Mandarin, in spontaneous speech, the segmentation and

Table 3. Please note that in case of unit-initial and -final discourse items, only those occur more than once are included in this table. The underlined, boldfaced items are relatively frequently used by the individual speakers.

	All discourse items	**Unit-initial discourse items**	**Unit-final discourse items**
1-A	A AI BA EI EN ER HAI HEI HONG HOU LA LE LEI MA NA NE O OU WA YA YO YOU	**A** **EI** **EN**	**A** E **EN** HONG **HOU** **LA** LE LEI MA NA NE **O** YA YO YOU
1-B	A EI EN HAI HEI HOU LA NE O YA YO YOU	**EI** **EN** HEI **HOU** O	**A** **EN** **HOU** **LA** NE O YA
2-C	A AI AIN E EIH EIHN HA HEIN HEN HO HON HUHN LA MA MHM MHMHM NA NHN OH ON OU UH UHM UHN WA YA	EIHN (2)	**A** **HO** **HON** **LA** NA OH OU UH YA
2-D	A AN EH EI EIH EIHN HA HAIN HEIHN HEIN HEN HO HON HUHN LA LAI LE MA MHM NA NE NEI NHN O OH OHN ON UH UHN YA	**A** OH **UH**	**A** HEIN **HON** **LA** NA **NE** UH YA
3-E	A EIHN HEIN HON LA MA MHM NA O YA YO	A HEIN	**A** LA NA **O**
3-F	A EIHN HEIN LA MA MHM MHMHM NUO O	**A**	**A** **MA**

verbalization of the so-called "unit of meaning" seem to be different from what we expect from a written sentence. In **Table 3**, it is clearly demonstrated that the particles *A* and *HOU* are used in both initial and final positions (unit-final occurrences are much more often than unit-initial ones). But *MA* and *LA* can only be used in final positions. It has to do with the discourse functions with which they are associated. *MA* and *LA* are usually referred to the whole utterance with an additional overtone.

We furthermore study discourse items in terms of their position within intonation units. **Table 4** summarizes the result. When discourse items are used in spontaneous speech, about 90% of them are found at the boundary irrespectively of speakers. This means that whenever a discourse item is produced, it either initiates or ends the verbalization unit of a new idea. Discourse items in spoken Mandarin mark the endings of intonation units much more often the beginnings.

Table 4. D-IUs refer to intonation units whose initial or final item is a discourse item. Monosyllabic D-IUs refer to intonation units composed of monosyllabic discourse items only.

Subjects	**1-A**	**1-B**	**2-C**	**2-D**	**3-E**	**3-F**
Total # of discourse items	215	148	131	200	74	40
# of monosyllabic D-IUs	36	31	35	40	9	8
# of unit-initial discourse items	18	36	10	21	9	10
# of unit-final discourse items	143	67	71	103	51	20
% of total D-IUs/all discourse items	92.1%	90.5%	89.3%	87%	93.2%	95%

3.2 Lexical Cues - Repair Resumption

To know whether repair structure interacts with the unit boundary, we have examined the re-initiation point, because where to resume a repair has been proved to be marked prosodically [12], therefore relevant to phrasing of speech production. **Table 5** gives the result in terms of the position of repair items within intonation units. More than half of the repair resumptions (re-initiation) are located in initial positions, even though speech repairs are one of the most typical spontaneous phenomena and are particularly irregular, as it is difficult to predict their location and form. This indicates that it is preferred to re-initiate an idea verbalization by means of a new intonation unit, despite unpredictable spontaneous language planning and production. The other interactions of repair structure and intonation units do not show such a clear tendency. Interestingly, 3-E and 3-F are young, fast-speaking, male speakers. Both of them clearly favor resuming a repair process with a new intonation unit (95% and 93%, respectively). Whether this has to do with sociolinguistic factors such as age or gender needs further investigation.

Table 5. Result of unit-initial repair resumption at the boundary

Subjects	**1-A**	**1-B**	**2-C**	**2-D**	**3-E**	**3-F**
Total # of repairs	96	24	39	50	52	28
# of resumed syllables in repairs	121	29	42	53	56	29
# of unit-initial repair resumption	69	18	29	37	53	27
% of unit-initial resumption	57%	62.1%	69.1%	69.8%	94.6%	93.1%

3.3 Temporal Cues - Articulation Rate and Unit Size

Examination of unit length sheds light on the speaker's preference and ability in speech production for verbalizing pieces of concepts. Chafe [2] reports for English an average length of 4 words per intonation unit. In Tao [18], the mean of length is 3-4 words per intonation unit. In our study, we chose syllables for the calculation. The reason is threefold. It is widely accepted that Taiwan Mandarin is a syllable-timed language. And in Taiwan Mandarin only grammatical particles are unstressed, so stress is not an essential contrast which needs to be considered to balance the prosodic weighting within words. Moreover, word segmentation is ambiguous for the Chinese writing system, as there are no blanks separating words in texts. Different native speakers may have different ways of segmenting words. As a result, **Table 6** lists the correlation between mean articulation rate and the unit size in syllables. It demonstrates a clear tendency: the faster a speaker speaks, the more syllables are produced in intonation units. That the relationship between the unit size and the articulation rate is stable implies that it is highly likely that intonation unit is a suitable unit for defining production units in Mandarin speech. This also shows that looking at the mean size is not enough, we need to take into account the speaking tempo, too.

We furthermore examine whether the speaking tempo changes consistently along different unit sizes. **Fig. 2** illustrates the relationship between the mean articulation rate of each unit and the unit size in syllables. Generally speaking, monosyllabic (for

Table 6. Unit size in syllables

Subject	1-A	1-B	2-C	2-D	3-E	3-F
Articulation rate (ms/ syllable)	175.7	195.7	193.3	184.3	154.6	159.7
Mean of IU size (# of syllables)	4.7	3.2	4.4	4.9	5.2	5.4
Median of IU size (# of syllables)	4	2	4	4	5	5

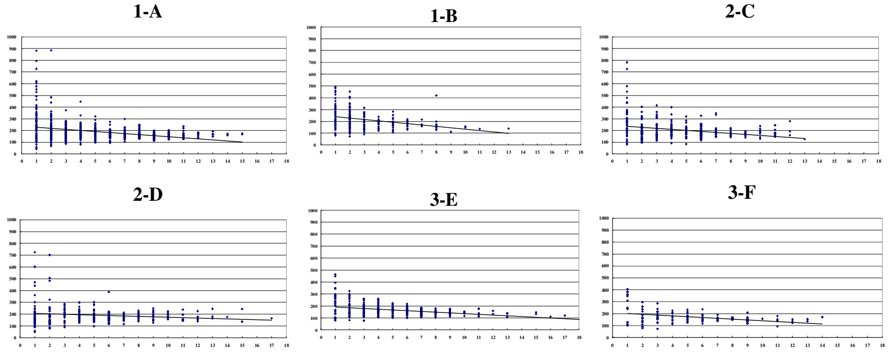


Fig. 2. The *x* axis is unit size in number of syllables, and the *y* axis is syllable duration means (msec) of the units of corresponding sizes. The syllable duration means decline, as the unit size increases.

all six speakers) and disyllabic (for five out of six speakers) intonation units have slower articulation rates, compared to those of other sizes. And the graphics show that when the unit size in syllables increases, the duration of each syllable in the unit decreases. The articulation rate and unit size in syllables correlate with each other consistently. This provides a further support for our choice of intonation unit as the segmentation unit and syllable as the measurement object.

3.4 Temporal Marking of Boundary Items

If intonation units are the units on which the language planning is based, then we should be able to find clear indication of their existence in speech production. As mentioned earlier, intonation units often end with a pause (40%-50% of the overall intonation units), and it is more likely that an intonation unit ends rather than starts with a discourse particle. To look for more indications of the existence of units, we calculated the syllable duration mean for all unit-initial and -final items. Unit-final syllables are significantly longer than unit-initial syllables for all speakers, as illustrated in **Fig. 3.** Compared with the overall syllable means, except speaker 1-B, all unit-initial syllable means are shorter and unit-final syllable means are longer. This temporal marking at boundaries seems to be a general and perhaps language-specific feature, independent of speakers and speaking situations.

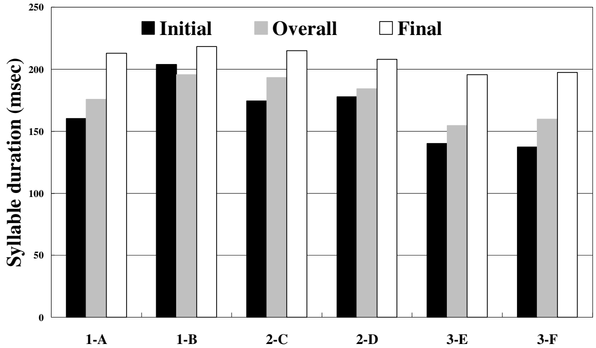


Fig. 3. Black, gray, and white bars are syllable means of unit-initial, overall, and unit-final syllables produced by each speakers. Unit-final items are significantly longer than the unit-initial items.

In addition to analysis of syllable mean applied on all data, we want to study whether the temporal marking at the boundary can be applied to each intonation unit. In conversation, a speaker may utter fast or slow "chunks" depending on the speaking situation and intention. We then calculated the syllable means of each units and compared them unit-wise with the corresponding unit-initial and –final syllable means. For this analysis, we applied one-way ANOVA to the duration of individual unit-initial and -final syllables dependent on the average syllable duration of each corresponding intonation units. **Table 7** shows a very clear tendency supporting the notion that regardless of the types of the boundary items, the unit-initial items are significantly shorter than the average speech rate of the intonation units they are located in; and the unit-final items are significantly longer than the average speech rate of the intonation units.

Table 7. Unit-wise temporal marking at boundary

Subjects	Unit-initial	Unit-final
1-A	F(1,486)=1.596; p=0.01	F(1,486)=2.964; p<0.001
1-B	F(1,137)=4.191; p=0.037	F(1,137)=17.115; p<0.001
2-C	F(1,265)=3.38; p<0.001	F(1,265)=5.799; p<0.001
2-D	F(1,234)=5.135; p<0.001	F(1,234)=5.172; p<0.001
3-E	F(1,320)=2.074; p=0.007	F(1,320)=1.884; p=0.015
3-F	F(1,132)=0.485; p=0.883	F(1,132)=0.731; p=0.742

Given an item, we also need to check the temporal relationship between its occurrences at the boundaries and the overall occurrences, i.e. item-wise comparison. Therefore, we analyzed all syllables (including lexical items and discourse items) which occur at the boundary more than once (i.e. either initial or final). For these items, two-tailed t-test was applied to the average duration of all the unit-initial occurrences and the average duration of the overall occurrences for all six speakers to test whether the difference is significant. The same process was also undertaken for the unit-final items. Because the data sample is not large enough, not all pairs are statistically significant. However, the tendency that unit-initial syllables are shorter than average and unit-final syllables are longer than average is clearly supported.

Table 8. Item-wise temporal marking at boundary

Subjects	Items (unit-initial)	Items (unit-final)
1-A	t(67)=-1.5; p=0.138	t(70)=4.181; p<0.001
1-B	t(27)=0.738; p=0.467	t(20)=0.85; p=0.405
2-C	t(38)=0.031; p=0.975	t(43)=1.446; p=0.155
2-D	t(42)=-0.684; p=0.498	t(30)=0.886; p=0.383
3-E	t(53)=-2.034; p=0.047	t(46)=5.659; p<0.001
3-F	t(24)=-2.959; p=0.01	t(29)=3.528; p<0.001

3.5 Temporal Characteristic of Discourse Items

This section analyzes the durational differences of discourse items at the boundary. As shown in **Fig. 4**, discourse items are normally longer than the overall syllables. The relationship between the overall syllable mean, the mean of discourse items, and the mean of syllables excluding discourse items is consistent across the speakers. Also, monosyllabic D-IUs are much longer than ordinary syllables and discourse items. These features may be language-specific, as all speakers behave themselves in a similar way. But such a consistency regarding the articulation rates of unit-initial and –final discourse items is not observed. In conversation, discourse items with different kinds of discourse functions and emphasis may be used in combination with differently varying speaking rates. This may be associated with the position regarding

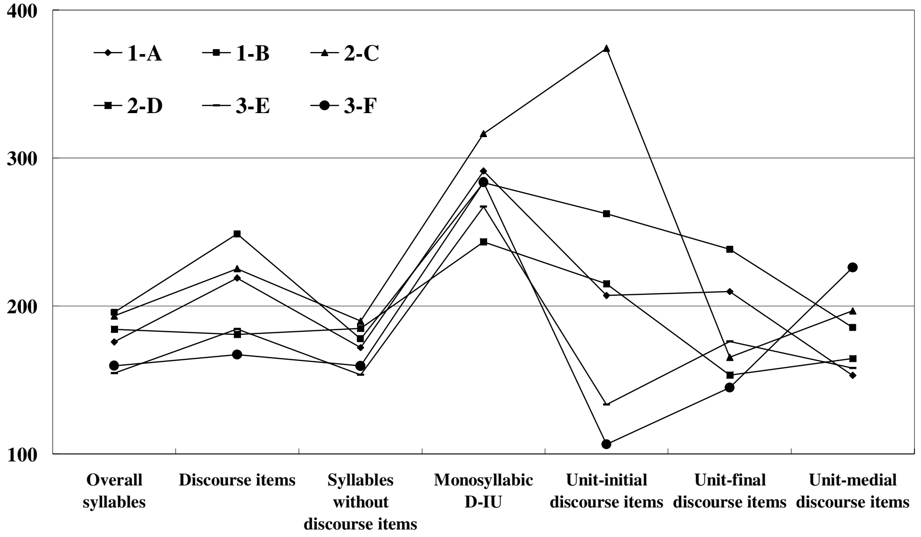


Fig. 4. The *y* axis represents the syllable duration mean (msec). It is observed that the behavior of the items to the left is similar across speakers, but those to the right are rather inconsistent.

intonation unit. This is illustrated in **Fig. 4**. Different degrees of tempo variability of discourse items at the boundary may be a speaker-specific feature, as the behavior across the speakers is very different.

4 Conclusion

Our study has shown that spontaneous Mandarin is segmented in terms of an intermediate unit between word and utterance, and it is likely that the unit is intonation unit. The fact that discourse items and repair resumption are preferably found at the boundary indicates that these lexical cues signal a kind of phrasing of spontaneous speech. Especially, discourse items mark important discourse locations, because more than 90% of them are located at intonation unit boundaries. Rallentando pattern seems to be a preferred rhythmic pattern for spoken Taiwan Mandarin, i.e. first fast, final slow pattern. And it is clearly found within the domain of intonation unit. The very different uses of speech tempo of discourse items indicate that the discourse functions associated with different kinds of discourse items may play an essential role. As a whole, this paper identifies the importance of intonation units in segmenting spontaneous speech, and works on finding more boundary cues such as intensity and pitch contours in terms of intonation units are in progress. For automatic speech recognition system, to first segment long speech data into smaller units like intonation units, then to retrieve the phonetic and semantic content of the units may be an effective strategy concerning spontaneous speech processing. This solution may help overcome the extreme difficulty of resolving the semantic content from the highly reduced phonetic information of spontaneous speech.

References

1. Chao, Y. R.: A Grammar of Spoken Chinese. Berkeley: University of California Press (1968)
2. Chafe, W.L.: Discourse, consciousness, and time: the flow and displacement of conscious experience in speaking and writing. University of Chicago Press (1994)
3. Clark, H. H., Wasow, T.: Repeating words in spontaneous speech, Cognitive Psychology, **37**, (1998) 201-242
4. Dankovicova, J.: Articulation rate variation within the intonation phrase in Czech and English. In: ICPhS 1999 (1999) 269-272
5. Dankovicova, J.: The domain of articulation rate variation in Czech, Journal of Phonetics, **25**, (1997) 287-312
6. Duanmu, S., Wakefield, G. H., Hsu, Y. P., Cristina, G., Qiu, S. P.: Taiwanese Putonghua Speech and Transcript Corpus, Linguistic Data Consortium (1998)
7. Du Bois, J., Schuetze-Coburn, S., Cumming, S., Paolino, D.: Outline of discourse transcription. In: Edwards, Lampert(eds.): Talking Data Transcription and Coding in Discourse Research (1993) 45-90
8. Fowler, C. A., Housum, J.: Talkers' signaling of "new" and "old" words in speech and listeners' perception and use of the distinction, Journal of Memory and Language, **26**, (1987) 489-504
9. Fujisaki, H.: Information, prosody, and modelling – with emphasis on tonal features of speech. In: Speech Prosody 2004. Invited keynote paper. (2004)
10. Hirst, D., Bouzon, C.: The effect of stress and boundaries on segmental duration in a corpus of authentic speech (British English). In: INTERSPEECH 2005, Lisbon (2005) 29-32.
11. Lee, T.-L., He, Y.-F., Huang, Y.-J., Tseng, S.-C., Eklund R. Prolongation in Spontaneous Mandarin. In: INTERSPEECH 2004-ICSLP, Jeju Island (2004) 526-529
12. Levelt, W. J., Cutler A.: Prosodic marking in speech repair, Journal of Semantics, **2**(2), (1983) 205-217
13. Li, C. N., Thompson S. A.: Mandarin Chinese: A Functional Reference Grammar. Berkeley: University of California Press (1981)
14. Low, E. L., Grabe E.: A contrastive study of prosody and lexical stress placement in Singapore English and British English, Language and Speech, **42**(1), (1999) 39-56
15. O'Shaughnessy, D.: Timing patterns in fluent and disfluent spontaneous speech. In: ICASSP 1995, Detroit (1995) 600-603
16. Oviatt, S., MacEachern, M., Levow, G.-A.: Predicting hyperarticulate speech during human-computer error resolution, Speech Communication, **24**(2), (1998) 87-110
17. Shriberg, E.: Acoustic Properties of Disfluent Repetitions. In: Proceedings of the International Congress of Phonetic Sciences, Stockholm (1995) 384-387
18. Tao, H.-Y.: Units in Mandarin Conversation. Prosody, Discourse, and Grammar. John Benjamins Publishing Company (1996)
19. Tseng, S.-C.: Repairs in Mandarin conversation. Journal of Chinese Linguistics, **34**(1), (2006) 80-120
20. Tseng, S.-C., Gibbon, D.: Discourse functions of duration in Mandarin: resource design and implementation. In: Proceedings of LREC (2006)
21. Zellner, B.: Pauses and the temporal structure of speech. In: Keller (ed.): Funfamentals of speech synthesis and speech recognition (1994) 41-62

Phonetic and Phonological Analysis of Focal Accents of Disyllabic Words in Standard Chinese

Yuan Jia[1], Ziyu Xiong[2], and Aijun Li[2]

[1] English Department, Nankai Univerisity
[2] Institute of Linguitics, Chinese Aademy of Social Sciences
jiayuan_nankai@126.com, xiongzy@cass.org.cn, liaj@cass.org.cn

Abstract. The article investigates the phonetic and phonological property of focal accents conveyed by disyllabic focused words with various tonal combinations in Standard Chinese. Phonetically, the effect of focal accents upon f_0 resides in two aspects: the manner and the condition of focal accents. Phonologically, the distribution of focal accents is mainly concerned. Acoustic and perceptual experiments and the underlying tonal target of focused constituents are employed in both phonetic realization and phonological analysis. Major findings are that: f_0 ranges of focused words are expanded as the H tones of both focused syllables are raised; the f_0 of the post-focus syllables are compressed obviously in the way the H tones of Tone1 and Tone2 are lowered; the realization of accents is closely related to the tonal target of the focused words; specifically, accents influence the acoustic performances of tones; furthermore, the combination of H/L determines the distribution of accents.

Keywords: focal accents; phonetic realization; phonological analysis; tonal target.

1 Introduction

In previous literature on English focus, in the phonetic aspect, Xu [1] proposed that narrow focus is realized by expanding the pitch range of the on-focus stress syllables and by suppressing the pitch range of the post-focus syllables while leaving the pitch range of the pre-focus syllables intact. In the phonological aspect, Ladd has highlighted the relation between focus and intonational nucleus placement [2] [3]. To this relation, Ladd adds the role played by abstract prominence patterns: the nucleus signals the focus of the utterance, and the nucleus is assigned to the element that bears the sentence stress in the sentence-level prominence pattern.

Studies of focus in Standard Chinese concentrate on the phonetic realization of focused constituents. Xu [4] investigates the formation of f_0 under the influence of focus by examining short Mandarin sentences with systematically varied tonal components and focuses. Results of his experiment demonstrate that the f_0 range is expanded by focus: the high points of H tones are raised while the low points of L tones are lowered. Further, the f_0 of all the words following the focus are substantially lowered no matter whether the f_0 of the focused words are raised or lowered. In contrast

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 55 – 66, 2006.

with the on-focus raising and post-focus lowering, the f_0 of the pre-focus syllables are barely changed.

Although the existing studies have reported changes in f_0 triggered by focus, phonetic and phonological explorations on the Chinese disyllabic words as a whole entities bearing focus, especially on the underlying factors constraining the distribution of these focal accents, have still been far less than those on English focus, and even absent at all. In this regard, we intend to examine the phonetic and phonological property of focal accents conveyed by Chinese disyllabic focused words with exhaustively various tonal combinations, and to offer a description of the effect of focal accents upon f_0 by the very means to reduce the tonal combination to tones in terms of H and/or L. We further endeavor to disclose the underlying forces constraining the distribution of the accent patterns.

2 Methodology

2.1 Experiment Designing

Focused words adopted in the acoustic and perceptual experiments are all disyllabic and with the morphosyntactic structure represented as Modifier-Head[1], or MH for short. Sixteen kinds of tonal combinations for the focused word of MH structure emerge from the formula of "[Tone1+Tone1] … [Tone4+Tone4]". And the composing syllables fall into a couple of sets: {Gan1, Tang2, Biao3, Da4} as the set of the first syllables, and {Ge1, Yi2, Jie3, Mei4} the set of the second. A certain focused word is thus constituted by the combination of each member from the first set and each from the second. All these words are set in four patterns of target sentences: 1) Jin1 Tian1 + [*focused words*] + Fei1 Dong1 Jing1; 2) Jin1 Chen2 + [*focused words*] + Hui2 Dong1 Jing1; 3) Jin1 Wan3 + [*focused words*] + Fan3 Dong1 Jing1; 4) Jin1 Ye4 + [*focused words*] + Qu4 Dong1 Jing1. It has been made apparent in previous studies that focal accents exert certain effect upon f_0 of the adjacent syllables. In the present study, however, the adjacent syllables are allocated with the four tones for the purpose of deliberating the effects imposed by the focal accents upon f_0 of these adjacent syllables under every tonal circumstance. The tones of the syllables apart from the focused words and the adjacent syllables are set invariably in Tone1. To approach to the different types of focus, we employ guide sentences which are mainly composed of *wh*-question equivalents. The guide sentence for all broad focus expressions in this article is invariably the interrogative of "Fa1 Sheng1 Le0 Shen2 Me0 Shi4? (What happened?)" And for narrow focus sentences we employ the guide sentences generally formulated as "Jin1 Tian1/Chen2/Wan3/Ye4 *Shei2* Fei1/Hui2/Fan3/Qu4 Dong1 Jing1? (*Who* fly to/go back to/go back to/go to Tokyo today/this morning/this evening/tonight?)".

2.2 Experiment Procedures

Four Standard Chinese speakers, one female and three males, ranging from 20 to 45 of age, were invited as the subjects. The recording was conducted in the sound-treated booth at the Institute of Linguistics, Chinese Academy of Social Sciences. The subjects

[1] Phonetic realization of focused words is different due to the different morphosyntactic structure of focused words (Jia, Xiong and Li) [5].

were seated comfortably in front of the computer screen, and the microphone was placed a hand away from mouth. During the recording, each sentence appeared on the screen three times in random order; meanwhile, the guide sentences were broadcast and the subject was asked to read aloud the displayed sentences as the response to the question in normal speed without any irregular pause. Sounds were recorded and saved directly into computer through sound recording software as "*wav*" file.

The target sentences in which the clause-middle focused words have 16 kinds of tonal combinations pronounced by a male and a female speaker were chosen for perceptual experiment samples. As has been mentioned above, each target sentence was recorded three times; only one was collected for each target sentence so as to reduce the unnecessarily huge amount of data. Since in this experiment the subjects are only required to give judgments over the patterns of the focused words in terms of *Weak* or *Strong*, and hardly any correlation exists between the circumferential tones and the focused words, thus, we render the tones both preceding and following the focused words as being Tone1. Totally three females and five males were invited to participate in the perceptual experiment. They are all standard Chinese speakers and show rather sensitive and stable perception. In order to minimize the negative analogical effect and perceptual fickleness, all the sentences involved were broadcast in sheer random order through perceptual software, and each sentence was perceived by as many repetitions as the subjects required in order to confirm their judgment. Thus, for each focused word we had two target sentences to provide to the subjects for judgment, and finally we got 8×2=16 samples for the word with the same tonal combination. Each subject was asked to finish the experiment individually, without any interpersonal consultation. During the experiment, four options appeared with the icons worded as: "the first syllable strong[2]", "the second syllable strong", "both strong" and "both not strong". The subject was expected to choose one of these options according to their decisive judgment of the weight of the focused words in target sentences.

2.3 Data Labeling and Extraction

Speech was first labeled by automatic segmentation software, and then the syllable boundaries were modified by hand. Before extracting the data, the manual refinement of the pitch tier was conducted in order to ensure the accuracy of the data. The data were retrieved by praat script with each syllable for 10 points, and the duration of the utterances was normalized. Finally, SPSS10.0 was used to get the means of f_0 for each target sentence.

3 Phonetic Realization of Focal Accents

In this part, two aspects of the effect of focal accents upon f_0 will be discussed: 1) the manner of effect, specifically taken to mean *rising* or *lowering* the f_0, and 2) the condition of effect, which means whether the lowering or rising imposes the effect on *H* or *L* tones. The fundamental unit we apply to describe these effects is the underlying tonal target, H or L; specifically, we render each focused syllable separately to divide

[2] Since each focused words in our experiment consists of a couple of syllables, it is sensible to require the subjects to tell which of the two syllables or whether both are strong, or not strong.

it into its original tones and in this manner to observe the performance, rather than deal with a single syllable as the descriptive unit.

The following two figures are the mean f_0 curves of sentences: “Jin1 Tian1 [*Gan1*] × [*Ge1, Yi2, Jie3, Mei4*] Fei1 Dong1 Jing1”, focused words [*Gan1*] × [*Ge1, Yi2, Jie3, Mei4*] are under both broad and narrow focus conditions. These sentences are adopted for the comparison among the pitch range changes of the second syllables of the focused words.

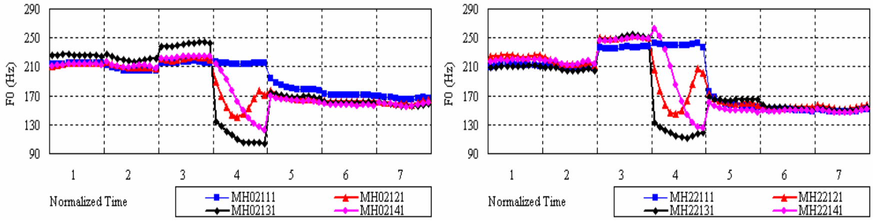


Fig. 1& 2. f_0 means of “Jin1 Tian1 [*Gan1*]×[*Ge1, Yi2, Jie3, Mei4*] Fei1 Dong1 Jing1” [Broad and Narrow Focus Conditions]

MH indicates that the morphosyntactic structure of focused word is Modifier-Head, “0” means the focus type is broad and “2” is narrow focus, the following “2” illustrates the position of the focused word is middle, the following “11” demonstrates the tonal combination is Tone1+Tone1 and “12” is Tone1+Tone2, etc, the last “1, 2, 3 and 4” show the tones adjacent to the focused word are Tone1, Tone2, Tone3 and Tone4.

Comparison of the two graphs shows that the pitch range of the second syllable is expanded by the focal accent, from the specific values, the pitch range of the broad focus syllable is 104Hz-215Hz and the difference of 215 minus 104 equals to 111Hz, while this difference of the second syllable range under narrow focus condition presents 137 Hz, which demonstrates that the pitch range of the second syllable of the focused words is expanded by the focal accent for 25Hz.

The following two graphs read “Jin1 Tian1 [*Gan1*, *Tang2*, *Biao3*, *Da4*]×[*Ge1*] Fei1 Dong1 Jing1” under both broad and narrow focus conditions from which the changes of pitch range of the first focused syllables can be seen.

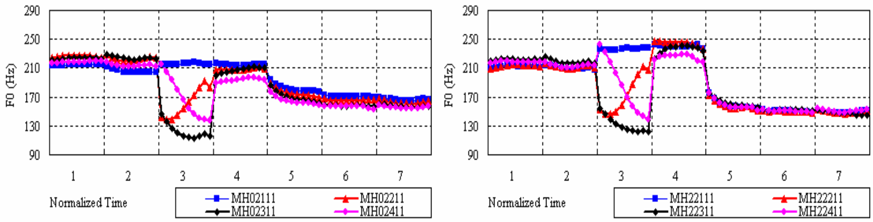


Fig. 3 & 4. f_0 means of “Jin1 Tian1 [*Gan1*, *Tang2*, *Biao3*, *Da4*]×[*Ge1*] Fei1 Dong1 Jing1” [Broad and Narrow Focus Conditions]

The above two figures illustrate that the pitch ranges of the first syllables are expanded by the focal accent, specifically, the pitch ranges of the first syllables of the broad focus words are within 114Hz-218Hz, and the pitch ranges of the narrow focused syllables are 122Hz-240Hz, meaning that the pith range of the first syllable is enlarged by 14Hz.

The above paragraph demonstrates that the pitch range is expanded by focal accents and the internal cause for the pitch range changes is analyzed through the direct comparison of the focused words under different focus conditions in the following figures. The following figure is the mean f_0 curves of sentence: "Jin1 Tian1 *Gan1 Ge1/Gan Mei4* Fei1 Dong1 Jing1", focused word "*Gan1 Ge1*" and "*Gan1 Mei4*" are under both narrow and broad focus conditions.

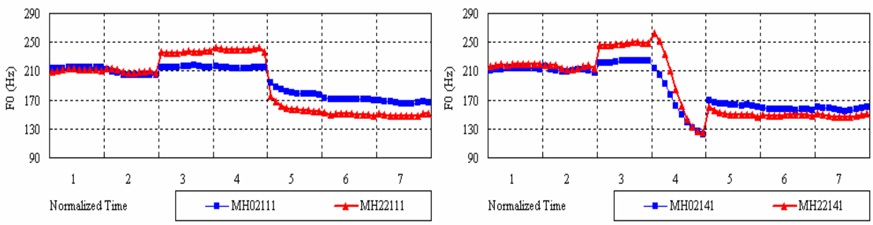


Fig. 5 & 6. f_0 means of "Jin1 Tian1 *Gan1 Ge1/Gan1 Mei4* Fei1 Dong1 Jing1"

Examination of the above two figures reveals that focal accents lift the f_0 of the H tones of the two syllables of the focused words while the L tone of the focused syllable resembles that of the broad focused syllable. The realization of focal accents manifests on the two focused syllables differs from English that the raising of the pitch is on the stressed syllable at the sentential level.

In general, the manners and conditions of the effect of focal accents upon f_0 in under-focus domain can be generalized in three aspects: firstly, the pitch range is expanded by the focal accent; secondly, the essential cause for the expansion of the f_0 range is the increase of the f_0 of the H tones, which indicates that the effect imposed by focal accents upon pitch mainly manifests itself on the H tones; thirdly, the effect exerted by focal accent on L tones is by no means obvious and systematic; fourthly, the manifestation of the effect of the focal accent on *both* of the focused syllables characterizes the feature as being language-specific of Chinese.

Figure 7 and 8 illustrate the effect of focal accents on f_0 range of the syllables in pre-focus and post-focus domain. Target sentences in Figure 7 are under broad focus atmosphere while Figure 8 under narrow one. The model of target sentences presents: "Jin1 Tian1 (Chen2/Wan3/Ye4)+[*Gan1 Ge1*]+ Fei1 (Hui2/Fan3/Qu4) Dong1 Jing1". As the pitch changes imposed by focal accents are under direct discussion, the tones of the focused words are set as Tone1.

Figure 7 and 8 demonstrate that the pitch range of the syllables following the focused word is compressed remarkably by focal accents while the pitch range of the syllables preceding the focused word remain much the same.

The intrinsic cause for the compressing of the pitch range of post-focus syllables can be obtained from Figure 5 and the following two graphs. The follow figures are the f_0 curves of "Jin1 Chen2/Ye4 *Gan1 Ge1* Hui2/Qu4 Dong1 Jing1".

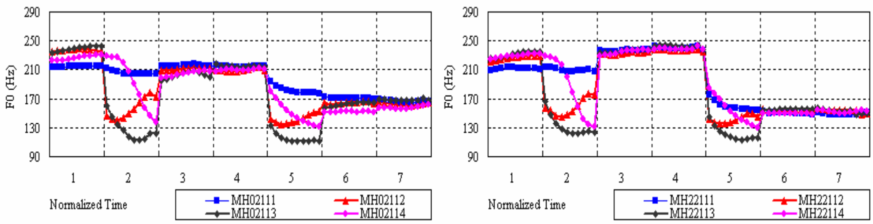


Fig. 7 & 8. f_0 means of "Jin1 Tian1 (Chen2/Wan3/Ye4)+[*Gan1 Ge1*]+Fei1(Hui2/Fan3/Qu4) Dong1 Jing1"

[Broad and Narrow Focus Conditions]

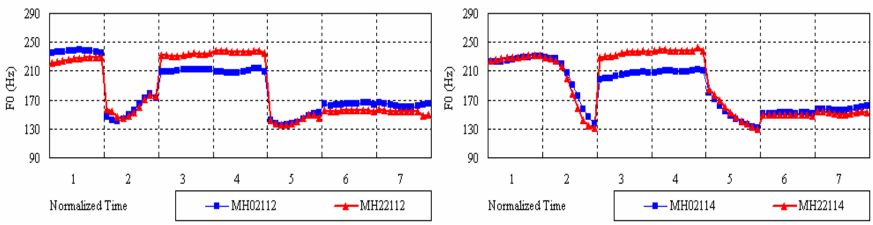


Fig. 9 & 10. f_0 means of "Jin1 Chen2\Ye4 *Gan1 Ge1* Hui2\Qu4 Dong1 Jing1"

Generally in the above mentioned three graphs have been discussed the phonetic realization of the focused word "*Gan1 Ge1*" with the differently assigned tones in surrounding. What is unanimously represented among the four figures is that f_0 in the focus position behave in the same manner, i.e., the f_0 of the H tones are remarkably raised under the effect of the focal accents. Pre-focus f_0 of the four tones, irrespective of whether being H or L tones, show either no perceptible difference or only slight disparity that is too erratic to be of any statistical significance. Post-focus f_0, however of the H tones of Tone1 and Tone2 are significantly compressed while those of the H tone of Tone4 show a minor height difference over that under broad focus condition, the reason why the H tone of Tone4 are not affected obviously is that the H tone locates itself in the transition part, and contains the L tone. All the L tones are hardly subject to the effect of the focal accents, with only quite negligible changes that also lack statistical importance.

4 Phonological Analysis of Focal Accents

Within the frame of Structure-based FTA[3] theory (Ladd & Gussenhoven)[2][6], the linguistic description of accent patterns involves two complementary but essentially separate aspects: a statement about which parts of an utterance are focused, and a statement about how a given pattern of focus is determined by the location of the accent. They argue that the speaker's decision about what to focus is subject to all kinds of contextual

[3] Gussenhoven proposes the "Focus-to-Accent" (FTA) approach. In very general terms, the FTA theory is that words and constituents in utterances can be focused for various reasons, and that focused words and constituents are marked by pitch accents.

influences that are difficult to identify. They also notice that once the focused part of the utterance is specified, the accent pattern follows more or less automatically by language-specific rules or structural principles. Therefore, in this part, the accent patterns of focused words at the sentential level with various tonal combinations will be described through the theoretical model of Metrical Phonology [7] based on the data obtained from the perceptual and experiments, further, the language-specific rules for restricting the distribution of the focal accents will be analyzed.

Table 1 demonstrates the perceptual results of accent patterns for each focused words.

Table 1. Perceptual results of focused words

Tonal Combinations	Both not strong	Both strong	First Strong	Second Strong	Total
Tone1+Tone1		3		13	16
Tone1+Tone2		2	11	3	16
Tone1+Tone3	1	4	7	4	16
Tone1+Tone4		6	8	2	16
Tone2+Tone1		5		11	16
Tone2+Tone2		6	2	8	16
Tone2+Tone3	2	4	6	4	16
Tone2+Tone4		6	1	9	16
Tone3+Tone1		3		13	16
Tone3+Tone2	1	3	2	10	16
Tone3+Tone3		4	7	5	16
Tone3+Tone4		5	1	10	16
Tone4+Tone1		5	3	8	16
Tone4+Tone2	1	6		9	16
Tone4+Tone3		1	10	5	16
Tone4+Tone4		5	1	10	16

It is can be seen from Table 1 that the first syllables of 8 focused words with the tonal combination of "Tone1+Tone4" were judged by the subjects to be "strong", which occupies 50% of the total number 16, whereas the "both strong" 37.5% and "second strong" 12.5%. In light of the phonetic realization shown by Figure 6, the accent pattern of "Tone1+Tone4" displays "s-w" relationship. Similar method can be adopted to approach the accent patterns of the other focused words with different tonal combinations, specifically, the accent patterns of the focused words of "Tone1+Tone2", "Tone1+Tone3", "Tone1+Tone4", "Tone2+Tone3", and "Tone3+Tone3" reflect "s-w" relationship; while the tonal combinations of "Tone2+Tone1", "Tone3+Tone1", "Tone3+Tone2", "Tone3+Tone4", "Tone4+Tone1", and "Tone4+Tone3" show the opposite accent relationship of "w-s". The remaining five tonal combinations, "Tone1+Tone1", "Tone2+Tone2", "Tone2+Tone4", "Tone4+Tone2", and "Tone4+Tone4", invariably show the “w-s” pattern through the perceptual experiment. The explanation for this strong tendency is to be made in Table 6.

The following tables show the accents distribution pattern of 16 different tonal combinations of MH structure from the word level to the sentential level through metrical grids (in the grids below, Line 0 stands for syllabic level, Line 1 for lexical level, and Line 2, sentential, “()” indicates syllable boundary):

Table 2. Phonological representation of [*Gan1*]×[*Ge1*, *Yi2*, *Jie3*, *Mei4*]

	Gan1	*Ge1*		Gan1	Yi2		Gan1	Jie3		Gan1	Mei4
Line2		×	Line2	×		Line2	×		Line2	×	
Line1	×	×	Line1	×	×	Line1	×	×	Line1	×	×
Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)

Lin [8] investigates the stress pattern of Mandarin disyllabic words in citation forms through perceptual experiment, and proposes that when disyllabic words are in normal stress, there is no absolute fixed stress pattern. This indicates that each syllable of MH structure words can bear word stress in citation forms.

As can be seen from the above table, the sentential accent dwells on the second syllable “Ge1” in “Gan1 Ge1”, and the accents dwell on the first syllable in “Gan1 Yi2”, the first syllable in “Gan1 Jie3”, and the first syllable in “Gan1 Mei4”.

Table 3. Phonological representation of [*Tang2*]×[*Ge1*, *Yi2*, *Jie3*, *Mei4*]

	Tang2	Ge1		*Tang2*	*Yi2*		Tang2	Jie3		*Tang2*	*Mei4*
Line2		×	Line2		×	Line2	×		Line2		×
Line1	×	×	Line1	×	×	Line1	×	×	Line1	×	×
Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)

And the table above shows that the syllables on which the sentential accents occur are “Ge1” in “Tang2 Ge1”, “Yi2” in “Tang2 Yi2”, “Tang2” in “Tang2 Jie3”, and “Mei4” in “Tang2 Mei4”.

Table 4. Phonological representation of [*Biao3*]×[*Ge1*, *Yi2, Jie3*, *Mei4*]

	Biao3	Ge1		Biao3	Yi2		Biao3	Jie3		Biao3	Mei4
Line2		×	Line2		×	Line2	×		Line2		×
Line1	×	×	Line1	×	×	Line1	×	×	Line1	×	×
Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)

From the above table can be seen that “Ge1” in “Biao3 Ge1”, “Yi2” in “Biao3 Yi2”, “Biao3” in “Biao3 Jie3”, and “Mei4” in “Biao3 Mei4” are the four syllables on which the sentential accents are located.

Table 5. Phonological representation of [*Da4*]×[*Ge1*, *Yi2*, *Jie3*, *Mei4*]

	Da4	Ge1		*Da4*	*Yi2*		Da4	Jie3		*Da4*	*Mei4*
Line2		×	Line2		×	Line2	×		Line2		×
Line1	×	×	Line1	×	×	Line1	×	×	Line1	×	×
Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)	Line0	(s)	(s)

And this table shows that the focal accents occupy the four positions of “Ge1” in “Da4 Ge1”, “Yi2” in “Da4 Yi2”, “Da4” in “Da4 Jie3”, and “Mei4” in “Da4 Mei4”.

We would like to explore the more unified explanation for the underlying causes for the shifting of the stress and the specific distribution patterns shown by the tables above. Sharing the very morphosyntactic structure of MH, the accent distribution patterns of these words, however, vary with the various tonal combinations. The pair of “Tang2 Ge1” and “Tang2 Jie3”, for instance, gives the accent patterns of “strong-final” and “strong-initial” respectively. The cause for the shifting of the stress from the first syllables to the second at the sentential level can seemingly be reduced to the simple fact that with the first syllable being the same in both words the only difference between them lies in the tones of the second syllables, Tone1 and Tone3 respectively. The accent distribution of above minimal pair of “Tang2 Ge1” and “Tang2 Jie3” implies that the explanations can be sought if the traditionally termed four tones have been reduced to the *original* tones marked duly by the very permutations of H and L, that is, Tone1 is transformed into HH, Tone2 LH, Tone3 LL, and Tone4, HL. Concretely, therefore, in “Tang2 Ge1” the syllable “Ge1” claims the accent just because “Ge1” can be re-rendered as “HH”, compared with “Tang2” as “LH”. “HH” overtakes “LH” in number of “H” and *therefore* appears strong. Similar situation holds true for “Tang2 Jie3” in which “H” in the “LH” pattern of “Tang2” outnumbers that of “LL” of “Jie3” and thus claims the accent. This correlation can be attested for the patterns of the other words listed below: “Strong-initial” patterns are found for “Gan1 Yi2”, “Gan1 Jie3”, “Gan1 Mei4” and “Da4 Jie3” because they can be re-written respectively as “HH LH”, “HH LL”, “HH HL” and “HL LL”; another “Strong-initial” pattern notably exists for “Biao3 Jie3”, which can be attributed to the tone sandhi of “Biao3 Jie3 → Biao2 Jie3”, that is, “LH LL”. Correspondingly, “Strong-final” patterns occur for “Biao3 Ge1”, “Biao3 Yi2”, “Biao3 Mei4” and “Da4 Ge1”, with the respective renderings of “LL HH”, “LL LH”, “LL HL” and “HL HH”.

However, a seemingly disturbing case emerges for five words in each of which the combination pattern in terms of “H/L” for the first syllables is totally *identical* in number of H to the corresponding second ones, as in “Gan1 Ge1” re-rendered as “HH HH”, “Tang2 Yi2” as “LH LH”, “Da4 Mei4” as “HL HL”, “Tang2 Mei4” as “LH HL”, and “Da4 Yi2” as “HL LH”. Solely from the perspective of “H/L” pattern the above five words should all have the pattern in which the first and the second syllables are *equally* strong. The fact both revealed by the perceptual experiment as well as phonetic graphs[4], however, turns out to be that it is the *second* syllables of the five words that bear the sentential accents and they fall into the “strong-final” pattern.

Therefore, by permutation, then, three patterns of contrast in terms of the amounts of H tones come into being. And they are, of the two syllables: 1) the first with more H

[4] As has been discussed above, the raising of f_0 is mainly manifested on the H tones.

tones, 2) the second with more H tones, and 3) each containing equally numbered H tones. The result of "more Hs claiming strong" can be explained through the phonetic analysis of focal accents that the realization of the focal accents mainly manifested on the H tones of the focused syllables. What, then, accounts for the pattern with "H/L" pattern being equal? Pierrehumbert [9] proposes that when two stressed syllables sound equal in pitch, the second is actually lower and when the perception of pitch is equal in height the second syllable bears more stress. In view of this, the height of the pitch of the H tones is the crucial factor accounting for the accent patterns of the word with equal number of H tones.

The following table provides the mean maximum pitch values of the H tones of the focused words under narrow focus condition with the tonal combinations of Tone1+Tone1, Tone2+Tone2, Tone2+Tone4, Tone4+Tone2 and Tone4+Tone4 (H1 indicates the mean pitch values of the H tones of the first focused syllable and H2 is the second, the unit for these values is Hz).

Table 6. Maximum pitch values of H tones

Tonal Combination	Tonal Feature	H1 (Hz)	H2 (Hz)
T1+T1	HH + HH	237	242
T2+T2	LH + LH	222	231
T2+T4	LH + HL	179	265
T4+T2	HL + LH	244	242
T4+T4	HL + HL	242	240

The extraction of the above values of f_0 under the circumstances of focus is achieved by smoothing off the influences exerted by the tones preceding and following the focused words, and thus by obtaining the means of the 16 focused words, each of which contains 48 samples.

The puzzling fact can be explained from the above values that the mean maximum pitch values of the second focused syllable are higher than the first ones in the tonal combinations of Tone1+Tone1, Tone2+Tone2 and Tone2+Tone4. And in the tonal combinations of Tone4+Tone2 and Tone4+Tone4 the heights of the H tones of the two syllables nearly equals each other. All these data demonstrate that the second syllables of the focused words of these tonal combinations bear stronger accent.

Therefore, in this part, we provide the phonological description of focused word with various tonal combinations through the theoretical model of Metrical Phonology. From the representation of the accent pattern of these focused words we offer the unified explanation for the underlying causes for the distribution of the accents by the means of re-rendering the tones of each syllable of focused word into the original H/L tones or the combinations of these two tones. When the underlying tonal combinations are different (this difference lies in the tonal combinations of each syllable between the focused words), the determining role of restricting the distribution of the accents is attributed to the amount of the H tones; however, when the underlying tonal combinations are identical, a predominant tendency exists of being always "strong-final". These results strongly indicate that the phonological composition of the focused syllable is the primary cause for restricting the distribution of the accents.

5 Conclusion

In this study, an integrated work of phonetic and phonological analysis for focal accents conveyed by disyllabic words in Standard Chinese has largely been done. Major findings are achieved from all the above analyses are expressed in both the phonetic and phonological aspects: for the phonetic perspective we have: 1) in the focus position under narrow focus condition, the pitch range of f_0 is enlarged as the H tones under narrow focus condition is significantly higher than under broad focus condition, and L tones show on obvious differences in f_0 under narrow focus condition, if there were, from those under broad focus condition, but the differences are *by no means* systematic and worthy of note. This finding, however, differs from that made by Yi Xu [4], who argues that under focus conditions f_0 of H tones becomes higher and of L tones, lower. 2) In the positions preceding the focus, f_0 range remains much the same, whether H tones or L tones, f_0 is but slightly subject to the effect exerted by the focus, and this finding bears no considerable divergence from that of Xu [4]. 3) Extremely significant effects imposed by the focal accents are found on the post-focus pitch range of syllables, which slightly but notably differs from the finding made by Xu [4] that the pitch range of f_0 of the post-focus syllables is considerably compressed by the focus. Our study, however, offers a more refined discovery that the focal accents do lower the f_0 of the H tones of Tone1 and Tone2, but exert only insignificant effects on the f_0 of the L tones of Tone2, Tone3 and Tone4 and of the H tones of Tone4. For phonological perspective, the underlying causes for the distribution of the focal accent are explored through the phonological description of the accent patterns of the focused words. The manifestation and distribution of focal accents are closely related with the underlying tonal target of the focused words that the realization of the tones is influenced by accents and the phonological composition of the focused constituents determines the distribution of the focal accents. We would therefore agree with Ladd [3] that once focus is identified in accordance with the speaker's intent and with the context, it is the language-specific structural factors that determine the accent distribution. In Standard Chinese, the phonological combination of the focused constituents is the primary causes that determine the distribution of the focal accents, this analysis also identical with Chomsky [10] that phonological characteristics determine the nature of the accents.

Acknowledgements

This research is supported by CASS key projects "Fundamental Speech Corpus and Pitch Pattern of Standard Chinese" YZDN50-05050 and "Research on Phonetics and Spontaneous Speech Processing".

References

1. Xu, Yi. 2005. Phonetic realization of focus in English declarative intonation. *Journal of Phonetics.*
2. Ladd, D. Robert. 1980. *The structure of Intonational Meaning: evidence from English.* Bloomington, Indiana: Indiana University Press.

3. Ladd, D. Robert. 1996. *Intonational phonology*. Cambridge University Press.
4. Xu, Yi. 1999. Effects of tone and focus on the formation and alignment of F_0 contours. *Journal of Phonetics*.
5. Jia, Yuan, Xiong Ziyu, Li Aijun. 2006. Narrow focus patterns of disyllabic words with different morphosyntactic structures in Standard Chinese. *TAL 2006 LA ROCHELLE* (France).
6. Gussenhoven, Carlos. 1983. Focus, mode and nucleus. *Journal of Linguistics* 19:377-417.
7. Liberman, Mark and Alan Prince. 1977. *On the Stress and Linguistic Rhythm*. LI 8:249-336.
8. Lin, Maocan. 1984. *Qingzhong Yin.* In Wu Zongji and Lin Maocan (eds.), *Shiyan Yuyinxue Gaiyao* (*An introduction to experimental phonetics*). Beijing: Commercial Press.
9. Pierrehumbert, Janet. 1979. The perception of fundamental frequency declination. *Journal of the Acoustical Society of America*, 66(2): 363-369.
10. Chomsky, Noam. 1971. Deep structure, surface structure, and semantic interpretation. In D. Steinberd and L. Jakobovits (eds.) *Semantics-an Interdisciplinary Reader in Philosophy*, *Linguistic and Psychology*, 183-216. Cambridge: Cambridge University Press.

Focus, Lexical Stress and Boundary Tone: Interaction of Three Prosodic Features

Lu Zhang[1], Yi-Qing Zu[2], and Run-Qiang Yan[3]

[1] School of International Studies, Zhejiang University, Hangzhou, 310058
luzhang@zju.edu.cn
[2] Motorola China Research Center, Shanghai, 200041
Yiqing.Zu@motorola.com
[3] Department of Biomedical Engineering,
Shanghai Jiaotong University, Shanghai 200030
yan_runqiang@hotmail.com

Abstract. This paper studies how focus, lexical stress and rising boundary tone act on F0 of the last preboundary word. We find that when the word is non focused, the rising boundary tone takes control almost from the beginning of the word and flattens F0 peak of the lexical stress. When the word is focused, the rising boundary tone is only dominant after F0 peak of lexical stress is formed. This peak is even higher than F0 height required by the rising boundary tone at the end of the word. Furthermore, the location of lexical stress restrains the height at F0 peak and high end to be reached. The interaction of these three factors on a single word leads to F0 competition due to limited articulatory dimensions. The study helps to build prosodic model for high quality speech synthesis.

Keywords: boundary tone, focus, lexical stress.

1 Introduction

Pitch contour is the acoustic manifestation of intonation. Fundamental frequency (F0) is the physical parameter to describe this variation. F0 is also used to convey linguistic distinctions, such as surd/voicing, tones in tone languages, lexical stress in stress languages; prosodic features, such as breaking and prominence; and paralinguistic features, such as the affective information denoted by the speaker. As they are encoded in the same F0, all these features have to compete and compromise to a certain extent in the manifestation of their functions, and this is one reason for the complexity of intonation.

The research of intonation basically concerns break and focus [3][4] [10][14[15][17][19]. Break is signaled by F0 changes of certain type right before the break. It reflects prosodic hierarchy, i.e. sentence, prosodic phrase, prosodic word, and foot. At certain levels, break carries a boundary tone, which uses a section of F0 right before the break. Focus directs attention towards the important or the new in an utterance [6]. In most cases F0 develops a peak to illustrate prominence (cf. low accent in [15]). Both boundary tone and focus can be competitive candidates in dominating F0, which can be judged by the height to be reached.

The variation of intonation leads to different sentence types. Declaratives typically have a gradual decline in F0 from the beginning, and questions usually adopt a rising

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 67 – 75, 2006.

intonation contour. [1][11][18] demonstrated that contrast between statements and questions could be seen clearly the more to the rightward of the sentence. Some researchers [21] concluded the difference between the two types existed in a longer temporal domain: F0 of the entire question was raised. Other researchers [8][13] reckoned the basic distinction of the two sentence types was due to terminal F0 movement determined by the word at the end of the sentence and the boundary tone of the prosodic unit. However, some experiments suggested the final rise for questions was not obligatory. In [7], for example, only about half of the questions had rising contours in the survey of yes-no questions in radio and television programs.

Besides boundary tones, many studies suggested that focus was also involved in the shaping of pitch contour [2][20]. In [5][9][12][16], for example, the relation of focus location or focus F0 height and sentence final F0 track were discussed.

To summarize briefly, most studies agreed that the basic distinction between statements and questions were concerned with boundary tones, usually falling for statements and rising for questions. The competing effect between the boundary tone and focus changed the shape of F0 finals: lifting statements and reducing questions.

The matter becomes more complicated, when a third factor involves in the competence, lexical stress of the word. The location of lexical stress differs, F0 of this preboundary word changes accordingly.

This paper therefore studies how the boundary tone, focus and lexical stress act on F0 based on the words that carry the rising boundary tone from a large English synthesis corpus. The study results provide useful information to prosodic control of high quality speech synthesis.

2 Methods

2.1 Corpus and Speaker

The speaker was a male American in his thirties. The recording was done in a sound-treated booth and the speaker was asked to read in a steady and natural way. The corpus contained 2640 sentences and was of the news broadcasting style.

2.2 Corpus Labeling

The corpus was labeled by two English majors who had phonetics training. The transcribers fully relied on their perception to annotate the corpus. Their task was to find boundaries in sentences, assign appropriate boundary tones and weigh the focused words in these sentences.

A break was labeled where boundary was sensed. The boundary was usually followed by obvious stops in the utterance.

Boundary tones fell into three categories, namely rising tone, falling tone and plateau. They were annotated according to what the labelers actually heard, after the place where a break existed was determined.

Focus of the sentence was assigned to those perceptually strong words.

2.3 Material Selection

In order to check how F0 was schemed to realize various features in the very limited articulatory dimensions, the final word before the boundary was selected. Besides the stressed syllable, the word of this type carried the major part of boundary tone, and might take focus of the sentence at the same time.

In this experiment, the final word before the boundary carrying a rising tone was chosen. Because we assumed, if the stressed information was usually presented by the pitch in the upper part of a speaker's frequency range, then the realization of falling track of this peak may be very interesting when there was also an urgent need to reach the high pitch target of the rising tone carried by the word itself. In a word, we would like to check how these three features were achieved in one single word: lexical stress, sentence focus and boundary tone.

Due to the restricted numbers of suitable four syllable words in the labeled corpus, we chose monosyllable, disyllable and trisyllable words as final materials. Besides, compound words were not included in the material, as to its twofold stresses in nature. The number of word in each word type could be found in Table 1, where "S" in word type refers to lexical stress. Thus, TriS2, for example, means trisyllable word with lexical stress on the second syllable.

Table 1. Word structures and material numbers. Mon, Di, and Tri means the word is with single, two and three syllable(s). S1, S2, S3 means the location of the word lexical stress is on the first, second and third syllable respectively.

Type	Mon	DiS1	DiS2	TriS1	TriS2	TriS3
Focused Word	67	117	12	83	33	0
Non Focused Word	235	186	40	55	32	0

2.4 F0 Measurement

The pitch contour of each word was inspected and the errors detected were manually corrected.

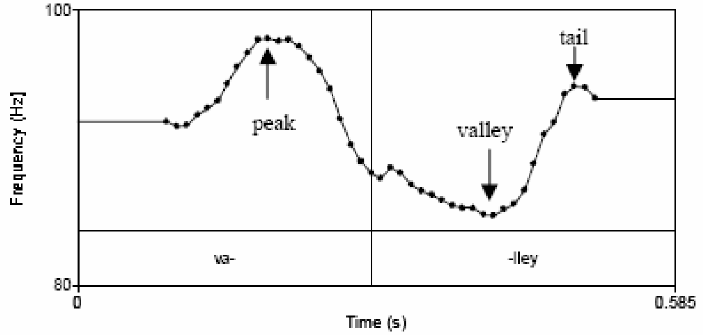


Fig. 1. Three measured points in the word "valley". Peak: the highest point of the stressed syllable; Valley: the lowest point after peak; Tail: the highest point after valley. The vertical line in the middle of the figure indicates the syllable boundary.

For each word, the pitch value at three points was measured. They were: the highest point of the stressed syllable (peak), the lowest point after peak (valley), and the highest point after valley (tail). Figure 1 takes the word "valley" as an example to show the measured points.

3 Results

In this part, di- and trisyllable words are compared. Monosyllable words would be discussed later in Section 4.2.

3.1 Non Focused Word

Theoretically speaking, peak, valley and tail are also the measured points for non focused category. However, after one-by-one check, we notice that word that is not carrying sentence stress does not have such a clear trajectory where obvious peak and valley could be defined. F0 movement of the stressed syllable is rather flat in this category, thus both peak and valley are actually nominal. In Table 2, peak is listed to prove that non focused word is with a tail higher than the highest point in the stressed syllable. Moreover, in DiS2, the nominal peak shares the same point with tail due to F0 shape of rising in general, thus peak is not measured in this type.

Table 2. Mean (M) and standard deviation (SD) of the value at peak and tail of non focused word

Type/Number	Peak		Tail	
	$M_{(Hz)}$	SD	$M_{(Hz)}$	SD
DiS1 /186	85.9	4.7	95.4	8.1
DiS2 /40	/	/	96.8	11.0
TriS1 /55	85.4	5.0	95.9	8.0
TriS2 /32	84.7	4.8	96.1	8.0

The analysis of the figures in Table 2 shows that tail in non focused word is higher than the nominal peak (DiS1, t=19.85; TriS1, t=10.65; TriS2, t=8.65; p<.05), while the height of tail in four categories is the same ($F_{(2, 270)}$=0.18, p>.05).

Valley is not listed in Table 2, because the location of valley varies. 165 out of 186 in DiS1, 23 out of 55 in TriS1 are in the first syllable. The remaining samples in these two types take rising from the second syllable. Besides, the lowest point after peak in all TriS2 samples is in the lexical stress, however, this valley is not necessary the start of rising. Further investigation shows that 22 out of 33 words in TriS2 take the rising from the first syllable, instead of the stressed second syllable.

3.2 Focused Word

Compared with their non focused counterparts, those sentence stresses have salient peaks, though they are carrying a rising boundary tone at the same time. Statistics show if this preboundary word is a sentence focus, the peak of the stressed syllable would have a higher value than its rising tail (DiS1, t=19.72; DiS2, t=7.72; TriS1, t=14.78; TriS2, t=13.78; p<.05).

Table 3. Mean (M) and standard deviation (SD) of the value at measured points of focused words

Type/ Number	Peak		Valley		Tail		Peak-Valley		Tail-Valley	
	$M_{(Hz)}$	SD	$M_{(Hz)}$	SD	$M_{(Hz)}$	SD	$M_{(Hz)}$	SD	$M_{(Hz)}$	SD
Di_S1 /117	118.4	13.0	87.0	5.1	97.5	7.3	31.4	13.1	10.5	5.4
Tri_S1 /83	116.0	14.7	83.0	5.8	95.2	8.1	33.0	15.1	12.2	6.8
Tri_S2 /33	114.6	9.3	88.8	5.4	96.7	7.2	25.7	7.1	7.8	4.0
Di_S2 /12	114.9	11.2	90.5	5.2	98.0	7.5	24.4	9.4	7.5	4.6

Table 4. Multiple Comparisons with valley, peak-valley, tail-valley as dependent variables. 1, 2, 3, 4 in (I) type and (J) type represents DiS1, TriS1, TriS2, DiS2 respectively. When test of homogeneity of variances shows significancy, Tamhane test is applied instead of Bonferroni. The figure in bold means the mean difference is significant at the .05 level.

(I) TYPE	(J) TYPE	Valley (Bonferroni)			Peak-Valley (Tamhane)			Tail-Valley (Tamhane)		
		Mean Diff (I-J)	Std. Error	Sig.	Mean Diff (I-J)	Std. Error	Sig.	Mean Diff (I-J)	Std. Error	Sig.
1	2	**4.0**	0.8	0.00	-1.6	2.1	0.97	-1.7	0.9	0.30
	3	-1.8	1.1	0.56	**5.7**	1.7	0.01	**2.7**	0.9	0.02
	4	-3.5	1.6	0.21	7.0	3.0	0.18	3.0	1.4	0.29
2	1	**-4.0**	0.8	0.00	1.6	2.1	0.97	1.7	0.9	0.30
	3	**-5.8**	1.1	0.00	**7.3**	2.1	0.00	**4.4**	1.0	0.00
	4	**-7.5**	1.7	0.00	8.6	3.2	0.08	**4.7**	1.5	0.04
3	1	1.8	1.1	0.56	**-5.7**	1.7	0.01	**-2.7**	0.9	0.02
	2	**5.8**	1.1	0.00	**-7.3**	2.1	0.00	**-4.4**	1.0	0.00
	4	-1.7	1.8	1.00	1.3	3.0	1.00	0.3	1.5	1.00
4	1	3.5	1.6	0.21	-7.0	3.0	0.18	-3.0	1.4	0.29
	2	**7.5**	1.7	0.00	-8.6	3.2	0.08	**-4.7**	1.5	0.04
	3	1.7	1.8	1.00	-1.3	3.0	1.00	-0.3	1.5	1.00

One-way ANOVA on mean F0 was conducted, with peak, valley, tail, peak-valley, tail-valley as dependent variables and word type as independent factor. The analysis of the statistics in Table 3 classifies these words into two categories: front stressed (DiS1 and TriS1) and back stressed (DiS2 and TriS2).

The four word types do not differ on peak ($F_{(3,\ 241)}$=1.10, p>.05) and tail ($F_{(3,\ 241)}$=1.64, p>.05). However, F0 height at valley is not even ($F_{(3,\ 241)}$= 15.82, p<.05). Post hoc test (Table 4) shows the lowest valley in all word types goes to TriS1. Meanwhile, detailed check shows the valleys of those TriS1 words mostly occur at the beginning of the last syllable (61 out of 83 cases), and the remaining occur later in the second syllable.

Compared with back stressed word, TriS1 climbs much higher from valley to tail ($F_{(3,\ 241)}$=5.92, p<.05). However, it does not differ from DiS1.

Another trisyllable word, TriS2, is lower both on relative peak (peak-valley) ($F_{(3,\ 241)}$=3.49, p<.05) and relative tail (tail-valley), compared with front stressed word.

4 Discussion

4.1 Di - and Trisyllable Words

Judging from the materials collected in our corpus, focus that carries a rising boundary tone would have a salient peak at lexical stress of the word. The value of peak is even far beyond the high rising tail at the end of the word. The rising tail has to compromise and develop a sharp movement only after the lexical stress reaches its peak. This result implies that in the competence of F0 movement required by the rising tone and the focus, the latter wins. The boundary tone carried by the word resumes its control after the peak, also suggesting the last syllable of the last word is the most important place for the completion of boundary tone. Figure 2 describes the trend of F0 tracks of focused words of four types with each syllable normalized to the same length.

In contrast, the peak of lexical stress of non focused word is blurred by the rising boundary tone. The location of valley in this type suggests most of the rising starts within the stressed syllable. Because the lexical stress is not prominent in the sentence as the one takes the sentence focus, in F0 formation of lexical stress and rising boundary tone, the latter takes the dominant position almost at the beginning of the word and makes lexical stress rather flat.

In details, TriS1 has the lowest valley. The reason might be that most of TriS1 valleys occur at the beginning of the last syllable (61 out of 83 cases). Thus while the first syllable of TriS1 is contributed to develop F0 peak, the word still has sufficient time to decline to the right point in later syllables and then complete the rising easily using at least the last syllable. In DiS1 and TriS2, though peak and valley forms in two separate syllables and the valley occurs much later than the one in TriS1, it does not guarantee a lower valley because a high end required by rising boundary tone is waiting. If the valley goes lower, the high tail of boundary tone can not be reached.

Meanwhile, TriS1 has a relative tail higher than back stressed word. However, it does not differ from DiS1. It can be inferred from the valley position of TriS1 that the rising tail of TriS1 is given comparatively sufficient time. Consequently, relative tail of DiS1 and TriS1 is the same, though at valley TriS1 is much lower.

For back stressed word, the height between valley and tail is compressed. In TriS2, both relative peak and relative tail are lower than front stressed word. The existence of the first syllable in TriS2 might postpone F0 peak in the stressed second syllable. The even more limited time restrains F0 to get down to a lower place and does not allow a higher tail. This can also explain the fact that DiS1 enjoys a higher relative tail than TriS2

Presumably, DiS2 would have lower relative peak and relative tail, as in DiS2 both peak and valley are in the same syllable. Theoretically, before an ideal peak could be generated, the pitch contour has to turn downward to prepare for the rise. However, the result is out of our expectation: DiS2 only differs from TriS1 statistically on relative tail. Nevertheless, we should notice this type only has 12 samples, which is quite limited in number. Further research is expected when enough samples are available, and we anticipate a different result.

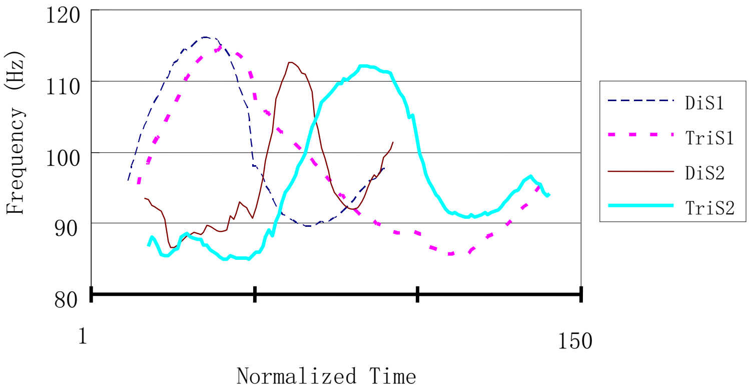


Fig. 2. F0 trend of focused word of four types. Each syllable is normalized to the same time period.

4.2 Monosyllable Word

While non focused monosyllables are just like other non focused types, having a rising contour in general, the focused monosyllable is very special. This type is originally supposed to be like DiS2, with peak and valley in one syllable. However, it behaves in a more diverse way. In all 67 samples, 16 are with an almost monotonous rising contour, and 20 has a lower peak than the tail, suggesting monosyllable words to be characteristically different at some place. One reason for so many variances might be that the single syllable word is very sensitive to the previous syllable. This is also a point that needs further studies.

5 Conclusion

In general, F0 movement of the last preboundary word that carries a rising boundary tone behaves with certain patterns. Due to limited articulatory dimensions, lexical stress of the word, focus information and the rising boundary tone carried by the same word have to compete for its share of F0 on this single word. The dominant factor differs with focus as the switch. When the word is non focused, boundary tone controls F0 almost from the beginning of the word, leaving F0 peak of lexical stress rather flat. When the word bears focus, the information of lexical stress is in the superior position. The rising boundary tone is dominant after F0 peak of lexical stress is formed. This peak is even higher than F0 height at the end of the word required by the rising boundary tone. Furthermore, the location of lexical stress restrains this F0 peak and end to be reached. The moving of lexical stress to the right end of the word may elicit the compressing of peak-valley, tail-valley range, suggesting a less thoroughly realized peak and tail, thus focus and boundary tone. The results prove that, each prosodic feature has its own dimension of realization in F0, but the extension and intensity of its performance is negotiable and competitory.

References

1. Bolinger, D. L.: Intonation across Languages. In: Greenberg, J. P. (eds.), Universals of Human Language, Vol. 2. Phonology. Stanford University Press, Stanford, CA (1978) 471-523
2. Cooper, W. E., Eady, S. J., Mueller, P. R.: Acoustical Aspects of Contrastive Stress in Question-Answer Contexts. Journal of the Acoustical Society of America, 77 (1985) 2142-2156
3. Di Cristo, A., Jankowski, J.: Prosodic Organization and Phrasing after Focus in French. Proc. of The 14th International Congress of Phonetic Sciences, San Francisco, 2 (1999) 1565-1568
4. D'Imperio, M.: Focus and Tonal Structure in Neapolian Italian. Speech Communication, 33 (2001) 339-356
5. Eady, S. J., Cooper, W. E.: Speech Intonation and Focus Location in Matched Statements and Questions. Journal of the Acoustical Society of America, 80 (1986) 402-416
6. Groves, P. L.: http://www.arts.monash.edu.au/english/resources/Metre/Accent.htm (1999)
7. Lee, W. R.: A Point about the Rise-Endings and Fall-Endings of Yes-No Questions. In: Waugh, L. R., van Schooneveld, C. H. (eds), The Melody of Language Intonation and Prosody. University Park Press, Baltimore (1980) 165-168
8. Lin, M.: On Production and Perception of Boundary Tone in Chinese Intonation. Proc. of the International Symposium on Tonal Aspects of Languages: With Emphasis on Tone Languages, Beijing (2004) 125-129
9. Liu, F., Xu, Y.: Parallel Encoding of Focus and Interrogative Meaning in Mandarin Intonation. Phonetica, 62 (2005) 70-87
10. Liu, F., Xu, Y.: Underlying Targets of Initial Glides – Evidence from Focus-Related F0 Alignments in English. Proc. of The 15th International Congress of Phonetic Sciences, Barcelona (2003) 1887-1890
11. Ma, JK.-Y., Ciocca, V., Whitehill, T. L.: The Effects of Intonation Patterns on Lexical Tone Production in Cantonese. Proc. of the International Symposium on Tonal Aspects of Languages: With Emphasis on Tone Languages, Beijing (2004) 133-136
12. McRoberts, G. W., Studdert-Kennedy, M., Shankweiler, D. P.: The Role of Fundamental Frequency in Signaling Linguistic Stress and Affect: Evidence for a Dissociation. Perception & Psychophysics, 57 (2) (1995) 159-174
13. Pierrehumbert, J.: The Phonology and Phonetics of English Intonation. Ph.D Dissertation. MIT, Cambridge, MA (1980)
14. Rump, H. H., Collier, R.: Focus Conditions and the Prominence of Pitch-Accented Syllables. Language and Speech, 39 (1996) 1-17
15. Silverman, K., Beckman, M., Pitrelli, M., Ostendorf, C., Wightman, P., Price, P., Pierrehumbert, J., Hirschberg, J.: ToBI: A Standard for Labelling English Prosody. Proc. of the International Conference on Spoken Language Processing, Banff, Canada, 2 (1992) 867-870
16. Studdert-Kennedy, M., Hadding, K.: Auditory and Linguistic Processes in the Perception of Intonation Contours. Language and Speech, 16 (1973) 293-313
17. Surendran, D., Levow, G.-A., Xu, Y.: Tone Recognition in Mandarin Using Focus. Proc. of Interspeech 2005, Lisbon, Portugal (2005) 3301-3304
18. Thorsen, N. G.: A Study of the Perception of Sentence Intonation – Evidence from Danish. Journal of the Acoustical Society of America, 67 (1980) 1014-1030

19. Xu, Y., Xu, C. X., Sun, X.: On the Temporal Domain of Focus. Proc. of the International Conference on Speech Prosody 2004, Nara, Japan (2004) 81-84
20. Xu, Y., Xu, C. X.: Phonetic realization of focus in English Declarative Intonation. Journal of Phonetics, 33 (2005) 157-197
21. Yuan, J., Shih, C., Kochanski, G. P.: Comparison of Declarative and Interrogative Intonation in Chinese. Proc. of The 1st International Conference on Speech Prosody, Aix-en-Provence, France (2002) 711-714

A Robust Voice Activity Detection Based on Noise Eigenspace Projection

Dongwen Ying[1], Yu Shi[2], Frank Soong[2], Jianwu Dang[1], and Xugang Lu[1]

[1] Japan Advanced Institute of Science and Technology,
Nomi city, Ishikawa, Japan, 923-1292
{dongwen, jdang}@jaist.ac.jp
[2] Microsoft Research Asia, Beijing
{yushi, frankkps}@microsoft.com

Abstract. A robust voice activity detector (VAD) is expected to increase the accuracy of ASR in noisy environments. This study focuses on how to extract robust information for designing a robust VAD. To do so, we construct a noise eigenspace by the principal component analysis of the noise covariance matrix. Projecting noise speech onto the eigenspace, it is found that available information with higher SNR is generally located in the channels with smaller eigenvalues. According to this finding, the available components of the speech are obtained by sorting the noise eigenspace. Based on the extracted high-SNR components, we proposed a robust voice activity detector. The threshold for deciding the available channels is determined using a histogram method. A probability-weighted speech presence is used to increase the reliability of the VAD. The proposed VAD is evaluated using TIMIT database mixed with a number of noises. Experiments showed that our algorithm performs better than traditional VAD algorithms.

Keywords: Voice activity detection, Principal component analysis, Auto-segmentation, Local noise estimation.

1 Introduction

The performance of speech processing systems such as Automatic Speech Recognition (ASR) systems, speech enhancement and coding systems, suffers substantial degradations in noise environments. By applying a robust Voice Activity Detection (VAD) algorithm to those systems, their performances can be improved in the adverse environments. In clean conditions, the VAD systems using short-term energy or zero-crossing features work fairly well [1], but in noisy conditions, a traditional VAD is no longer robust when speech signal is seriously contaminated by noise. It is still a challenging problem to design a robust VAD for noise environments.

In the past twenty years, many researches have been conducted to obtain a robust VAD in adverse environments. Some of the researches paid attention to the intrinsic speech features such as periodic measure [2]. The other methods focused on the statistical model of speech and noise signals, such as the Gaussian statistical model based VAD [3] [4], Laplacian model based VAD [5] and high-order statistical VAD

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 76 – 86, 2006.

[6]. However, in low Signal-to-Noise Ratios (SNR) condition, speech features and speech statistical characteristics were not easy to be obtained. To reduce the noise effect, recently, a method combining speech enhancement with VAD was proposed [8]. Their method, however, has the two problems in the speech enhancement stage: residual noise and speech distortion, which brought error to VAD.

In this paper, we propose a novel approach to realize a robust VAD. The basic consideration is that speech usually has a different distribution from noises in the energy domain. If we can sort the components that have low power for noise and high power for speech, it is possible to extract more reliable information for speech even if the average SNR of the noisy speech is low. For this purpose, first, a noise eigenspace is constructed based on an estimated covariance matrix of noise observations using Principal Component Analysis (PCA). Projecting the noisy speech onto the noise eigenspace, the reliable information can be found out in the sub-eigenspace with smaller eigenvalues. Thus, a robust VAD can be realized based on the reliable information. Section 2 introduces the principles of noise eigenspace projection. Section 3 shows the implementation of the algorithm. In Section 4, we give the experimental evaluation, and compare our algorithm with some leading algorithms.

2 Projection in Noise Eigenspace

This section first investigates the SNR distribution property in a noise eigenspace. Then, we describe how to obtain the noise eigenspace in real application.

2.1 SNR Distribution in Noise Eigenspace

The noise eigenspace is used to describe the property of noise energy distribution. It is constructed from by principal component analysis of noise covariance matrix. Using eigenvalue decomposition, we can get the following relationship between eigenvalues and eigenvectors:

$$C\varphi_k = \lambda_k \varphi_k, \quad k = 1, 2, ..., K \tag{1}$$

where C is the covariance matrix of a zero mean noise signal n, $\varphi(k)$ is the eigenvector corresponding to eigenvalue λ_k. By sorting the eigen-coordinates based on eigenvalues order $\lambda_1 > \lambda_2 >, ..., > \lambda_K$, we get the corresponding eigenvectors $\{\varphi_k | k = 1,2,...,K\}$. The projection of a noisy speech frame x on the kth eigen-coordinate then is written as:

$$y_k = x \cdot \varphi_k \tag{2}$$

Since the noise energy centers on some coordinates, when projecting noisy speech into the noise eigenspace, it is possible to find a sub-eigenspace with few noise energy, hence higher SNR, where we can extract available information. Here, we use a specific noise to demonstrate the idea how to extract available information from noisy speech based on the noise eigenspace. We construct a noise eigenspace from a period of destroyer-engine noise. A speech sentence is mixed with the period of noise at 0dB. Both the speech and noise are respectively projected into the eigenspace. Since covariance matrix is calculated from the whole period of mixed noise, noise projection energy is actually the noise eigenvalue of the corresponding eigen-coordinate. The results of this processing are shown in Fig. 1. The left panel of Fig. 1 illustrates the initial distribution of projection energy in the original eigenspace. The blue curve is noise projection energy and the red is the projection energy of the clean speech. We sort eigenvalues in a descending order and rearrange the coordinate of the eigenspace according to the sorted order, where speech projections will move with the noise eigenvector in pair. For example, the channel with the maximum noise and the projected speech, shown by the dashed line in the left panel, are transferred to the lowest channel in the sorted noise eigenspace. Thus, a monotonically descending curve of the noise energy is obtained as shown in middle panel of Fig. 1, and the corresponding speech projections are shown in red curve with non-monotonic changes. In the rearranged space, one can see that in the high coordinates the speech's energy is higher than that of noise even though the average SNR is equal to zero or lower. Especially in last coordinates, the SNRs are much larger than the original SNR, as shown in right panel of Fig.1.

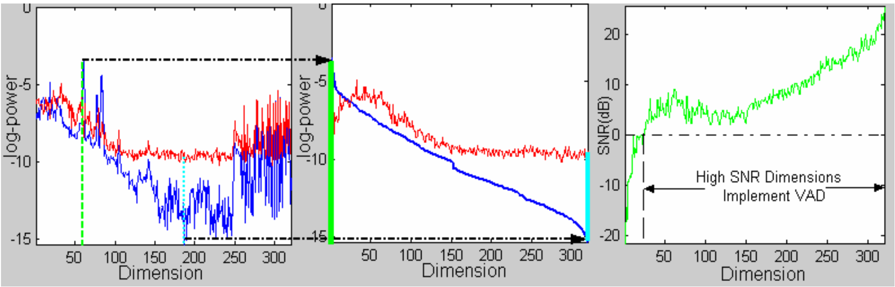


Fig. 1. Energy distributions in a noise eigenspace

For investigating the generality, the noisy speech projections are testified using eigenspaces of other types of noises out of the NOISEX'92 database. We mixed the noises with clean speech sentences from TIMIT database at given SNR levels. In real application, it's impossible to calculate the noise covariance matrix from the whole period of mixed noise. So, we estimate the covariance matrix by the non-speech period at each sentence beginning (as described in section 2.2).

Then, we project the noise and speech onto the sorted eigenspace and measure the SNR at each coordinate. Here we define the projection SNR ξ_i of the i$^{\text{th}}$ coordinate as the difference between the i$^{\text{th}}$ coordinate SNR and the mixture SNR, as described in formula (3):

$$\xi_i = 10\log_{10}(S_i / N_i) - 10\log_{10}(S / N) \tag{3}$$

where S and N are the total energy of a speech sentence and the mixed noise respectively. S_i and N_i are the projection energy of speech and noise at the ith coordinate respectively. The energy in the original space equals the summation of projected energy at each coordinate:

$$S = \sum_{k=1}^{K} S_k \quad \text{and} \quad N = \sum_{k=1}^{K} N_k$$

Thus, we further rewrite the formula as:

$$\xi_i = 10\log_{10}(S_i / \sum_{k=1}^{K} S_k) - 10\log_{10}(N_i / \sum_{k=1}^{K} N_k) \tag{4}$$

From formula, we can find out that projection SNR ξ_i is only concerned with the percentage of energy distribution at the ith coordinate. Since, projection SNR has no relationship with the global average SNR, we can easily represent the relationship among projection SNR, eigen-coordinate index and distribution probability by a three-dimension color image.

The color image is constructed by this way. For each sentence, we can calculate its projection SNR at each coordinate. At a given coordinate, we construct a histogram to describe the projection SNR distribution of all noisy sentences, and represent the value as probability of occurrence. So, the probability summation of each coordinate equals to 1. We combine the histograms at all coordinates into a colored image. In this algorithm, the speech sampling rate is 16 kHz, frame length 0.02s and frame shift 0.01s. Thus, the full eigenspace has 320 eigen-coordinates.

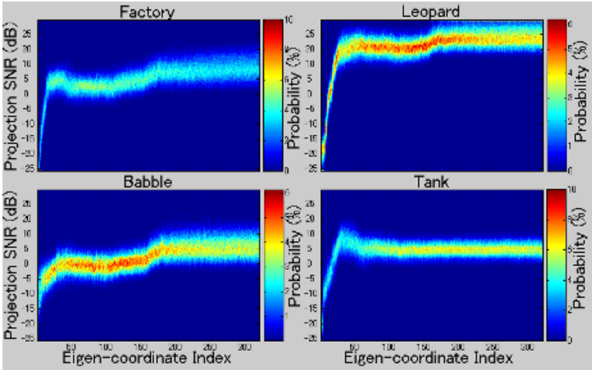


Fig. 2. Projection SNR distribution in noise eigenspace. Vertical axes describe the projection SNR. The color represents its distribution probability.

From the figure, it's easy to understand that the SNR of the projected signal on high dimensional coordinates is greater than that of projection on low dimensional coordinates. In another word, the SNR have an increasing tendency from the low to high coordinates. The statistics experiment shows the projections on eigen-coordinates with smaller eigenvalues always associate with high SNR. Therefore, it's possible to utilize the information of coordinates with smaller eigenvalues and ignore the coordinates with larger eigenvalues to carry out robust VAD.

2.2 Noise Eigenspace Estimation

Noise covariance matrix is the basis of eigenspace calculation. Before implementing VAD in eigenspace, it is necessary to obtain a reliable estimation of noise covariance matrix from noisy speech. Suppose there is somewhat a non-speech period in the beginning of each sentence, an initial covariance matrix can be estimated from this period. Then, the covariance matrix is updated stepwise using the detected noise.

To obtain a credible estimation of the initial noise covariance matrix, the frame shift is reduced to 0.375ms so that we can obtain 350 noise frames within 140ms at the beginning of sentences. The noise eigenspace is updated based on a time-varying estimation of the covariance matrix $\hat{C}(n)$ ($K \times K$). Giving an initial estimation $\hat{C}(0)$, it is successively updated as:

$$\hat{C}(n) = \alpha\hat{C}(n-1) + (1-\alpha)x(n)x^T(n) \tag{5}$$

where n is time (frame) index, α is a low-pass, forgetting factor with value 0.98, $x(n)$ is the observed noisy signal vector.

As known, eigenvalue decomposition is a time-consuming operation. Since noise is much more stationary comparing to speech signal, it's possible to doing eigenvalue decomposition periodically. On one hand, a longer period for eigenvalue decomposition can save computation time. On the other hand, a shorter period will benefit to an accurate estimation of noise eigenspace. So, a tradeoff is made between computation time and the accuracy of eigenspace.

3 Voice Activity Detection in Noise Eigenspace

In this section, we address how to detect the voice activity in the sub-eigenspace with high SNR. Before the noisy speech projected into noise eigenspace, the input signal is partitioned into homogenous segments as units for VAD decision. We construct

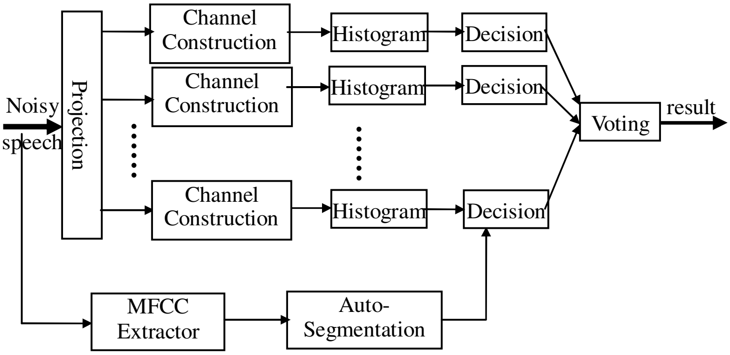


Fig. 3. Block diagram of the proposed VAD

channels using high-SNR coordinates and realize a sub-VAD at each channel. At last, the reliable channels with greater SNR will give a voting. The processing block diagram is shown in Fig. 3.

3.1 Auto-segmentation and Channel Construction

Firstly, we use auto-segmentation to partition the frame sequence into homogeneous segments. It is based on the consideration that, in noisy speech signal, the voiced and unvoiced blocks usually occur as segments consisting of several consecutive frames. The decision results should not transfer between speech and noise frame by frame. Here, homogeneous segments are taken as units for VAD decision, which reduces the problem of spurious changes of speech detection and limits speech-noise transfer times in the decision. The algorithm is a dynamic programming based procedure to minimize the segmentation cost [9]. In our algorithm, eight-dimension MFCC features including the log-power energy are used for auto-segmentation.

Secondly, the noisy speech frames are projected onto the noise eigenspace. Then, the every 10 adjacent projections are grouped into one channel by using the logarithm of the absolute magnitude summation to form a smoothed envelope. There are totally 32 grouped, projected channels in our algorithm.

The constructed channels located at the low dimensional coordinates have low SNR. Those channels bring much speech false alarm and contribute a little to speech hit rate. Therefore, those channels should be ignored in decision. Here, the channel SNR is used to evaluate each channel's reliability. It is estimated based on eigenvalues (average noise energy) and observed projection energy. According to experiments, the channels with SNR less than 2dB should be ignored in VAD decision. The left channels are used for VAD.

3.2 Histogram Based Local Noise Estimation

For making a correct final VAD decision, we carried out a sub-VAD decision at each channel. To do so, an appropriate threshold for each channel should be given. We propose a histogram-based method to estimate the sub-VAD threshold. The sub-VAD threshold is decided by noise level and variance of noise log-power. Suppose that the noise log-power of each channel obeys a Gaussian distribution, the problem arrived at estimation of the mean (noise level) and variance of the Gaussian function.

Many approaches such as clustering [9] and GMM fitting [7] have been proposed to estimate noise level in noisy speech. All these methods are based on the following observations in the histograms of log-power energy of noisy speech [10]:

a. In the two peak mode of the histogram, the peak in lower region is usually contributed by background noise, while the peak in lower region is contributed by speech.
b. In general, the noise mode has a salient peak and its variance is smaller than that of speech. The reason is that, as commonly assumed, the energy of the background noise is more stationary than that of speech.

c. The two modes are clearly separated in high SNR conditions. As SNR is decreasing, the two modes are getting closer and eventually merge into one mode.

However, in most situations, the two-peak mode assumption is not kept well. There may be only one peak model in speech pause duration or the mode with more than two peaks on the histogram. Traditional ways for estimation of noise level can not deal with those situations. It is necessary to design a local noise estimation method to deal with one peak, two peaks, and several peaks cases. Our estimation method only concern with noise mode, since noise mode is more salient than speech mode. Based on the basic observation in (a), (b) and (c), we present a local noise estimation method, as following steps:

i. Taking a dynamic range (0~9dB relative to minimum power) to construct the 40-bin histogram. This range is wide enough to include the noise level.
ii. Using a 3-point median filter to smooth the occurrence number, and taking the first peak at left side as the noise level location.

The noise level is the average of noise log-power Gaussian model. It is also assumed that noise log-power less than the noise level is affected little by speech as shown by shadow in Fig. 4. Then, its variance is estimated by the data less than the noise level. Based upon the local noise Gaussian model, we can define a sub-VAD threshold:

$$\text{Threshold} = \mu + \gamma\sigma \tag{6}$$

where μ is the noise level, σ is the estimated variance, γ is the coefficient for tuning the threshold. Fig. 5 illustrates the sub-VAD threshold estimation of noisy speech at 5dB in factory noise situation using the histogram method. The thick curve in the upper panel is noisy speech power envelope; the thin curve is clean speech power envelope. The dark segments in the middle panel are the detected speech segments. The threshold is calculated using formula (6). The centroids of homogenous segments

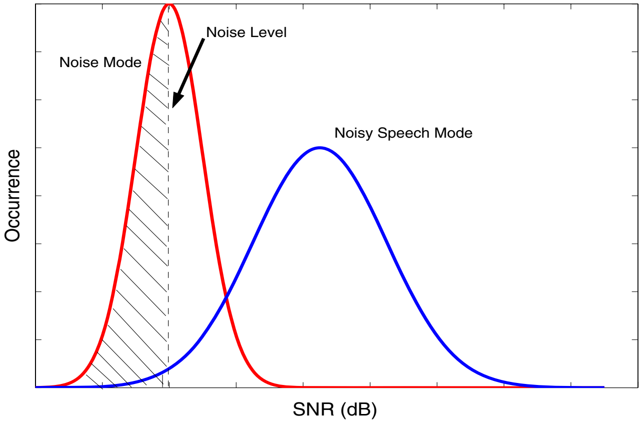


Fig. 4. Noise and noisy speech mode

partitioned by auto-segmentation are compared with sub-VAD threshold. Our local noise estimation method can deal with all cases, whether in speech pause, high or low SNR conditions.

The coefficient γ tuning the sub-VAD threshold in formula (4) should adapt to channel SNR. In high SNR channels, γ should be smaller to make the sub-VAD sensitive to speech and be larger in low SNR channels to avoid speech false alarm. According to experiments, when γ is linearly interpolated 1.3~1.1 between 2dB~8dB, it achieves better tradeoff between speech false alarm rate and hit rate. If channels' SNR is higher than 8dB, γ equals 1.1.

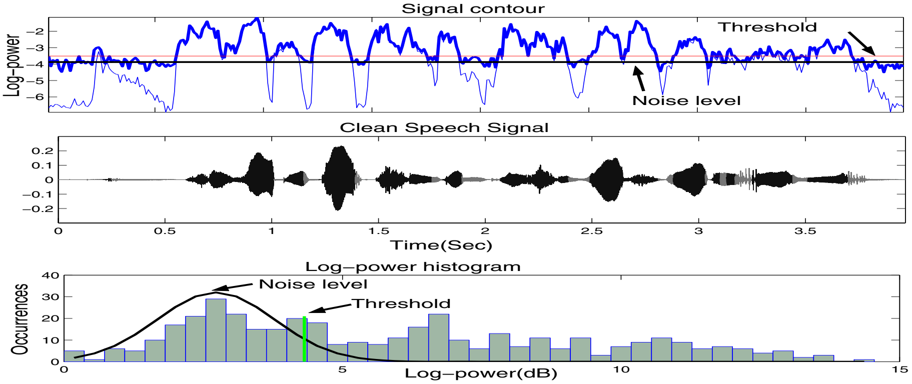


Fig. 5. Noise estimation by histogram

3.3 Voting and Parameter Adaptation

As mentioned in section 3.1, the channels with SNR less than 2dB are ignored. Only those channels with SNR larger than 2dB take part in the voting. So, the numbers of voting channels varies with average SNR conditions, it's necessary to normalize the votes by channel numbers. If the normalized votes exceed the threshold δ , the homogenous segments will be decided as speech. Fig. 6 is the voting result of a speech sentence mixed with babble noise at SNR=0dB. There are 30 channels with SNR larger than 2dB, taking part in the voting. In the middle panel, the red part is the detected speech segments.

Considering the tradeoff between noise and speech hit rate, in real application, we adapt the voting threshold δ to the average SNR level as:

$$\delta = \begin{cases} \delta_l & SNR < SNR_l \\ round(\dfrac{\delta_h - \delta_l}{SNR_h - SNR_l}(SNR - SNR_l) + \delta_l) & SNR_l < SNR < SNR_h \\ \delta_h & SNR > SNR_h \end{cases} \tag{7}$$

where SNR_h and SNR_l are the highest and lowest SNR levels respectively in real applications; δ_h and δ_l are the voting thresholds corresponding to the highest and

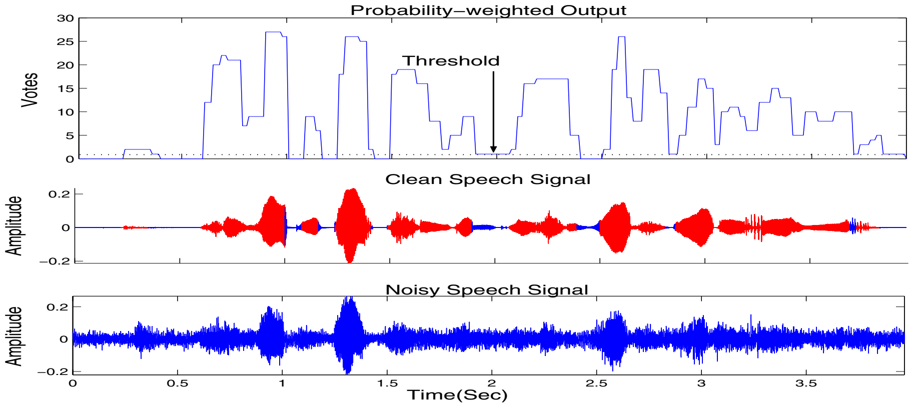


Fig. 6. Detection results of 0dB babble noise

lowest SNR levels. For SNR between lowest and highest levels, the voting threshold is linearly interpolated between δ_l and δ_h; *round* is the nearest integer function.

4 Experimental Evaluation

To evaluate the effectiveness of our VAD algorithm, we measured the detection probability (including speech hit rate HR1 and noise hit rate HR0) for a number of noisy speech paragraphs. The experiment data were taken from the TIMIT database. We connected every ten sentences from one speaker into a speech paragraph and mixed it with noise taken from NOISEX'92 database at variant SNR situations. Our experiment data consisted of 168 paragraphs with duration of about half a minute. The VAD references were labeled based on energy envelopes of clean speech signals.

In the detection, the paragraphs were chopped into 4-second segments. The noise eigenspace was estimated as described in section 2.2. For every 4 seconds, the noise eigenspace was updated by the detected noise. The adaptive voting threshold was calculated using formula (7), where the parameters were set as $\delta_l = 1$ for $SNR_l = -5dB$ and $\delta_h = 6$ for $SNR_l = 20dB$.

Table 1. Experimental results

		G.729B	AFE (Wiener Filtering)	Proposed VAD
Factory	HR1	77.91%	89.91%	94.77%
	HR0	84.43%	40.76%	58.48%
Babble	HR1	74.79%	86.43%	91.18%
	HR0	74.99%	45.30%	55.81%
Tank	HR1	77.21%	90.86%	94.62%
	HR0	85.25%	36.75%	64.74%

Table 1 shows the experiment results of our proposed algorithm with the traditional VAD algorithms. The values in the table are the noise hit rate (HR0) and speech hit rate (HR1) averaged over noisy speech different SNR from -5dB to 20dB. In this table, one can see that, in noisy environments, our algorithm works much better than G.729B [1] and AFE [11] algorithms.

5 Conclusions

In this paper, we proposed a noise eigenspace based VAD algorithm. A local noise estimation method was implemented in the proposed method to increase the robustness of the detection. The experiments showed that our algorithm were much more robust than traditional VAD algorithms, such as G.729 and AFE VAD algorithms.

Acknowledgements

Partial of this research was done when the first author worked for MSRA as an intern. It is supported by the program for the ”Fostering Talent in Emergent Research Fields” in Special Coordination Funds for Promoting Science and Technology by Ministry of Education, Culture, Sports, Science and Technology.

References

1. A Silence Compression Scheme for G.729 Optimized for Terminals Conforming to Recommendation V.70, 1996. ITU, ITU-T Rec. G.729-Annex B.
2. R. Tucker: Voice activity detection using a periodicity measure. Proc.Inst. Elect. Eng., 139(4):377-380, 1992.
3. Y. Ephraim and D. Malah: Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator. IEEE Trans. Acoustic., Speech, Signal processing, ASSP-32:1109-1121, 1984.
4. J. Sohn and W. Sung; A voice activity detector employing soft decision based noise spectrum adaptation. Proc. ICASSP, pp. 365-368, 1998.
5. S. Gazor and W. Zhang: A soft voice activity detector based on a Laplacian-Gaussian model. IEEE Trans. Speech Audio Process. 11(5): 498-505, 2003.
6. E. Nemer, R. Goubran, and S. Mahmoud; Robust voice activity detection using higher-order statistics in the lpc residual domain. IEEE Trans. Speech Audio Process. 9(3):217-231, 2001.
7. Q. Li, J. Zheng, A. Tsai, and Q. Zhou: Robust endpoint detection and energy normalization for real-time speech and speaker recognition. IEEE Trans. Speech Audio Process., 10(3):146-157, 2002.
8. J.Ramirez, J.C. Segura and et al.: An effective subband osf-based VAD with noise reduction for robust speech recognition. IEEE Trans. Speech Audio Process., 11(5):498-505, 2003.

9. Yu Shi, Frank K. Soong, and Jian-Lai Zhou: Auto-segmentation based partitioning and clustering approach to robust end pointing. Proc. ICASSP2006 .
10. Ris C, Dupont S.: Assessing local noise level estimation methods: application to noise robust ASR. Speech Communication, 34:141-158, 2001.
11. ETSI ES 2011 08 recommendation, 2000. Speech processing, transmission and quality aspects (STQ); distributed speech recognition; front-end feature extraction algorithm; compression algorithms.

Pitch Mean Based Frequency Warping

Jian Liu, Thomas Fang Zheng, and Wenhu Wu

Center for Speech Technology, Tsinghua National laboratory for Information Science and Technology, Tsinghua University, Beijing, 100084
liuj@cst.cs.tsinghua.edu.cn, {fzheng, wuwh}@tsinghua.edu.cn

Abstract. In this paper, a novel pitch mean based frequency warping (PMFW) method is proposed to reduce the pitch variability in speech signals at the front-end of speech recognition. The warp factors used in this process are calculated based on the average pitch of a speech segment. Two functions to describe the relations between the frequency warping factor and the pitch mean are defined and compared. We use a simple method to perform frequency warping in the Mel-filter bank frequencies based on different warping factors. To solve the problem of mismatch in bandwidth between the original and the warped spectra, the Mel-filters selection strategy is proposed. At last, the PMFW mel-frequency cepstral coefficient (MFCC) is extracted based on the regular MFCC with several modifications. Experimental results show that the new PMFW MFCCs are more distinctive than the regular MFCCs.

Keywords: Pitch, frequency warping, MFCC.

1 Introduction

State-of-the-art speech recognition systems have to face with a lot of variability in the acoustic signal. For example, context variability, style variability, speaker variability and environment variability are some typical types of variability [1] that may cause mismatch between training and test data of automatic speech recognition (ASR) systems. A lot of schemes have been developed in the past few years to compensate for this mismatch in order to improve the accuracy of the ASR system. Two major schemes are acoustic features transformation and acoustic model parameters adaptation.

Speaker variability and speaking style variability are two major factors that might cause mismatch between the trained acoustic models and the actual speech to be recognized. To reduce speaker variability, vocal tract length normalization [2, 3, 4] is commonly used to transform acoustic feature for ASR. The correlation between a speaker's average pitch and the vocal tract length was also exploited in [5]. On the other hand, even for a same speaker, his/her speech will change much with different speaking styles, therefore the speaking style variability also need to be considered. The pitch contour is one of the important features to classify different speaking style. There is a correlation between the pitch contour and the speaking style. Thus both the speaker variability and the speaking style variability correlate with the pitch frequency.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 87 – 94, 2006.

Automatic speech recognition should be based on speech features that contain relevant information capable of discriminating different speech sounds. The dynamic features of pitch are useful in speech recognition, especially for Chinese. Yet speech signals with different average pitches could contain the same phonetic information. Even the phase vocoder can manipulate the signal in frequency-domain, enabling pitch-shifting without changing the phonetic information [6, 7]. Thus the average pitch of speech is not the relevant feature to discriminate different phonetic information in speech for ASR. Commonly used speech features such as MFCCs are affected by changes of the pitch in speech signals. One way to alleviate the disturbance of pitch is to find speech features that are less sensitive to changes of pitch, yet capable of retaining good discriminative properties.

In this paper, a pitch mean based frequency warping (PMFW) method is proposed in feature extraction to compensate for the pitch-mismatch in speech signals. In [8, 9], the formant-based frequency warping was discussed for speaker normalization. However, the motivation of this paper is not only implementing speaker normalization. Because the average pitch is not directly proportional to the vocal tract length [5], using pitch explicitly for speaker normalization is not so reasonable as expected. On the other hand, the average pitch of the speech segment does have relations to both the speakers and the speaking styles. Effects of the pitch-mismatch need to be considered separately in ASR. Our work presents an approach that warps the frequency according to the average pitch of a speech segment. The motivation here is to integrate the PMFW into the acoustic feature extraction at the front-end with a little computation and make the new PMFW features more discriminative for ASR.

2 Pitch Mean Based Frequency Warping

2.1 ASR and Pitch

Pitch plays an important role in speech perception. The pitch is not a characteristic of the vocal tract length and does not directly affect the resonant frequencies. However, the information about pitch can be used to improve ASR systems. There are three typical methods that use the pitch information in ASR systems.

First, the pitch can be used as an acoustic feature and modeled using hidden Markov models (HMMs) and/or Artificial Neural Network (ANN). For example, in [10], the dependency between the hidden state and the pitch was modeled implicitly. The ASR system could achieve significant improvement by incorporating the pitch frequency.

Second, the pitch can be used to synchronize the frame size and/or the shift. A constant frame size and a constant shift are always used in ASR systems. The power spectral estimation may include artifacts without aligning the frames to the natural pitch cycles. A pseudo pitch synchronous method was proposed in [11] which improved the robustness and accuracy for low SNR speech.

Third, the pitch can be used for frequency warping factor estimation. In [5, 12], the correlation between a speaker's average pitch and the vocal tract length was exploited and the probability distribution of warp factors conditioned on pitch observations was modeled. That pitch-based warp factor estimation can be an effective method of improving ASR performance.

In ASR systems, the MFCC is one of major acoustic features. MFCC features are calculated from the power spectrum, and include some harmonic structure related to the pitch. Variations in pitch could cause variations in features. As a result the pitch mismatch and the variability in features have effects on speech recognition systems. On the other hand, the pitch variability and the speaker variability do not have direct relations. Even the same speaker could have pitch variability, such as at different mental conditions. The variability due to pitch will be implicitly alleviated by training a speech recognition system on a corpus collected from a large, diverse collection of speakers. However, the explicit reduction or elimination of pitch-included feature variability could lead to better recognition performance.

2.2 Pitch Mean Based Frequency Warping

Frequency warping is a typical kind of methods for feature transformation. The frequency axis is scaled by a warping function $f_\alpha(\omega)$, where α is a warping factor. Given the power spectrum, $X(\omega)$, of a speech signal, the warped spectrum is

$$Y(\omega) = X\left(f_\alpha(\omega)\right) \tag{1}$$

The warping function $f_\alpha(\omega)$ is always assumed invertible, i.e. strictly monotonic and continuous [3]. The warping function should conserve the bandwidth and information contained in the original spectrum in theory. However, there is redundant information in the original spectrum and only a subband of spectrum is useful for frequency warping. In our work, a linear frequency function is used, i.e. $f_\alpha(\omega) = \alpha\ \omega$. The reason for using a linear frequency function is that it has explicit physical meaning. According to the Fourier transformation, the compressing or stretching in frequency axis is equivalent to the re-sampling of the waveform in time axis, i.e. $X(\alpha\ \omega) = \frac{1}{\alpha} x(t/\alpha)$. Thus warping frequency with a linear function could alleviate the pitch-mismatch in the speech signal. Generally speaking, the phonetic information is '*hidden*' in the relative spectrum. The frequency warping adjusts the spectrum to determine more distinctive bands for ASR in some sense. It has been proved that the Maximum-Likelihood (ML) based frequency warping is effective for ASR [2], however, it requires more data and computation. Because perceiving pitch is natural for human and human can process speech properly with pitch variations, such as singing and speaking, we will focus on the relations between the pitch and the warping factor.

How to determine the warping factor α is important for frequency warping. In our method, the warping factor is dependent of the average pitch of a certain speech segment. We can assume that the warping factor α is a function of the average pitch as follows

$$\alpha = g(p) \tag{2}$$

where p is the pitch mean of a speech segment. Our goal here is to determine an analytic approach to expressing the relationship between the warping factor and the average pitch. Thus, two monotonic and continuous functions are exploited and compared in our experiments. Two typical functions are defined as follows

$$g(p) = \alpha_{\min} + (\alpha_{\max} - \alpha_{\min}) \bullet \frac{(p - p_{\min})}{(p_{\max} - p_{\min})} \tag{3}$$

$$g(p) = 1 + \frac{(\alpha_{\max} - \alpha_{\min})}{2} \bullet \frac{\log_2 \left(p^2/(p_{\min} \bullet p_{\max})\right)}{\log_2 \left(p_{\max}/p_{\min}\right)} \tag{4}$$

where $p_{\min}$ and $p_{\max}$ are the minimal and maximal pitch values of the human voice, respectively, $\alpha_{\min}$ and $\alpha_{\max}$ are the lowest and highest bounds of the warping factor, and p is the pitch mean of a speech segment. Empirically the pitch range of human voice is from 50 Hz to 500 Hz approximately. In our experiments, the range of the pitch is from 55 Hz to 440 Hz for convenience and any pitch with its value lower (or higher) than 55 (or 440) Hz is set to 55 (or 440) Hz. $\alpha_{\min}$ (or $\alpha_{\max}$) is also determined empirically as 0.85 (or 1.15) in our experiments. Equation (3) is in a linear form meaning that the warping factor is proportional to the average pitch in a linear space while Equation (4) is in a nonlinear form meaning that the warping factor is proportional to the average pitch in an octave space.

The pitch mean of the speech segment is calculated as

$$p = \frac{1}{N} \sum_{\substack{t=0 \\ 55 \le p_t \le 440}}^{t=T} p_t \tag{5}$$

where T is the total frame number of a speech segment, p_t is the pitch value at frame t (if no pitch value is successfully estimated at frame t, p_t is set to 0 in the above equation), and N is the total number of frames at which pitch value is successfully estimated. Note that T could be set as a fixed time period such as 2 seconds, however, for convenience, in our experiments, each sentence is considered as a speech segment and T is set to its length.

In practice, errors in pitch estimation are inevitable. However the average of pitch in a speech segment can be calculated with little bias if the speech segment is long enough. The method in [13] can be used to balance the doubling error rates and the halving error rates in pitch estimation to get more accuracy pitch mean in a speech segment.

2.3 PMFW Derived Feature

The proposed PMFW is integrated in the feature extraction at the front-end without additional computation in the training and the decoding procedures. The PMFW features are based on the standard MFCCs with two additional steps added as shown in Fig. 1.

First, the pitch mean based frequency warping is performed in the Mel-filters frequencies. The frequency warping can be implemented by simply varying the spacing and width of the component filters of the filter bank without changing the original speech signal [2]. The PMFW is implemented as follows

$$B'(k) = \alpha B(k), \quad k = 0,1,...,N+1 \tag{6}$$

where $B(k)$ is the start frequency of Mel filter k (for example, $B(0)$, $B(1)$, $B(2)$ are start, middle and end frequencies of the first Mel filter, respectively), and N is the total number of Mel filters. Equation (6) means that a male speaker with a smaller α would use a relative low band of frequency to calculate features, and vice versa. It can be assumed that male speakers' phonetic information is hidden in relative low frequency bands and female speakers' is hidden in relative high frequency bands.

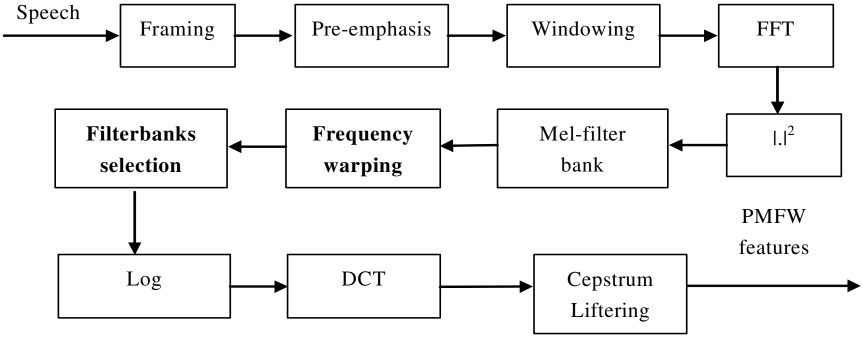


Fig. 1. Schematic diagram for PMFW feature extraction

Second, Mel filters are selected. Different warping factors will bring mismatch in bandwidth between different speech signals. In [2, 3], the piecewise warping functions were used to solve this problem. Using the piecewise warping function ensures that the full frequency band is used in features at different warping factors. However, the full band is not used in our method. We determine stable sub bands by cutting off a fixed number of lowest/highest bands at different warping factors. The selected sub bands should contain the same number of Mel filters. The number of filters that should be cut off is determined by

$$n = \underset{\alpha_{\max} B(k) \le f_{\max}}{\arg\max}\ \alpha_{\max} B(k) \tag{7}$$

where $f_{\max}$ denotes the maximal signal bandwidth. The filter start frequencies selected after PMFW are $B'(N+1-n)$, …, $B'(n)$. Both the lower and the higher filters will be cut off. For the experiments described in this paper, the sampling rate is fixed at 16 kHz, imposing a limit on the maximum signal bandwidth of 8 kHz. 35 (N=35) Mel filters are used and n=34 calculated by using Equation (7). Thus $B'(2)$, …, $B'(34)$ are selected to use for feature extraction after PMFW.

3 Experiments and Discussion

3.1 Experimental Setup

Experiments were designed to compare the performance of systems using traditional MFCCs and PMFW MFCCs. A subset of 863CSL corpus [14], which is a continuous speech database, was used in our experiments. The training set contained 20 male speakers' data and 20 female speakers' data, totally 21,749 sentences (10,824 sentences for male and 10,925 for females) for about 22 hours. The test set contained another 8 male speakers' data and 8 female speakers' data, totally 8,941 sentences (4,524 for males and 4,417 for females) for about 9 hours.

In our experiments, we used HTK version 3.2 [15] for training, testing, and the baseline's MFCCs feature extraction. The PMFW MFCCs were extracted by our own program using the algorithm proposed in this paper. The pitch, MFCCS and PMFW MFCCs were extracted every 12 milliseconds. Both PMFW MFCCs and traditional MFCCs were 26 dimensional, consisting of 13 static coefficients and corresponding

Table 1. Recognition results with gender matched/mismatched training and test data ('Linear' means using Equation (3) to calculate the warping factor while 'Octave' using Equation (4) to calculate the warping factor; *n* M (or *n* F) means the training or testing set contains speech data by *n* male (or female) speakers; the performance is evaluated in syllable accuracy rate, in %)

Training Set / Test Set	Method	20 M (%)	20 F (%)
8 M	Baseline	67.69	15.42
	Linear	69.90	48.59
	Octave	69.31	44.60
8 F	Baseline	20.71	78.90
	Linear	53.87	81.03
	Octave	48.78	80.56

Table 2. Recognition results with gender independent training (*n* M (or *n* F) means the training or testing set contains speech data by *n* male (or female) speakers; the performance is evaluated in syllable accuracy rate, in %))

Training Set / Test Set	Method	20 M +20 F (%)
8 M	Baseline	67.27
	Linear	70.99
	Octave	70.02
8 F	Baseline	78.04
	Linear	80.77
	Octave	80.19
8 M + 8 F	Baseline	72.59
	Linear	75.82
	Octave	75.04

13 delta coefficients. The 5-state 8-mixture Hidden Markov Model (HMM) topology was adopted to model the toneless tri-IFs where IF means either a Chinese initial or a Chinese final. The speech recognition units were 397 Chinese syllables (with tone disregarded).

3.2 Results and Discussion

The first experiment was designed to compare the performance between traditional MFCCs and the PMFW MFCCs on gender dependent models. Table 1 illustrates that when the traditional MFCCs are used, there will be very large drops in syllable accuracy rate when there is gender mismatch between the training speakers and the testing speakers. When the PMFW MFCCs are used, the accuracy rate will be improved considerably no matter the training speakers and the test speakers match in gender or not.

Traditional MFCCs perform badly when the training and test speakers gender mismatch. Thus some ASR systems use gender dependent models and perform gender recognition before speech recognition. The PMFW MFCCs could alleviate variations caused by gender mismatch. Although the accuracy rates in the gender mismatch test are lower than that in the gender matched test when PMFW MFCCs used, the accuracy will be remarkably increased in contrast to traditional MFCCs. Comparing two functions for calculating the warping factor, we can see that using a linear function to restrict the warping factor and the pitch mean could achieve higher accuracy at all tests in contrast to the octave function.

The second experiment was designed to compare the performance between traditional MFCCs and the PMFW MFCCs on the gender independent models. The percentages correct for the baseline when tested with 8 male and 8 female speakers were 67.27% and 78.04%, respectively, which were lower than those (67.69% and 78.90%, respectively) in the first experiment. The size of the training set in the second experiment was about twice larger than that in the first experiment, however, the accuracy rates were lower. The reason for that could be that traditional MFCCs of male and female speech are relatively diverged in the feature space, although the acoustic models used in two experiments both were 5-state 8-mixture based HMMs. Thus, it might be more difficult to model the distributions of the gender independent features than the gender dependent features when the acoustic model parameter size is fixed.

According to the first experiment, the linear function for warping factor calculation had better performance, so here we will only discuss the results when using the linear function here. When using PMFW MFCCs, the percentages correct when tested with 8 male and 8 female speakers were 70.99% and 80.77%, respectively, in the second experiment, and 69.90% and 81.03%, respectively, in the first experiment. It shows that the accuracy for males has been increased by 1.09%, while that for females has been decreased by 0.26%. Compared with traditional MFCCs, the PMFW MFCCs could have better performance, in other words, the PMFW MFCCs can have more convergence in the feature space than traditional MFCCs. Furthermore, in 16-speaker test (8 male and 8 female), there was a syllable error rate reduction of 11.8% when linear PMFW MFCCs were used in contrast to the traditional MFCCs.

4 Conclusion

The motivation of this paper is to extract more distinctive features at the front-end with little extra computation. By exploiting the correlation between the pitch and the speech, we propose an effective pitch mean based frequency warping method. To alleviate the pitch variations in speech signals, the warping factor is considered as a function of the average pitch of a speech segment. Then, two typical functions of the pitch mean are defined to calculate the warping factor. Furthermore, a simple method for performing frequency warping in the Mel-filter bank frequencies is described. The Mel filters selection strategy is presented for solving the mismatch in bandwidth between the original and the warped spectrum. Based on these operations, the PMFW MFCCs is extracted instead of the traditional MFCCs. Experimental results show that the PMFW MFCCs have better performance than traditional MFCCs

References

1. Huang, X. D., Acero A., Hon H. W.. *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*. Prentice Hall, New Jersey, 2001.
2. Lee L., Rose R. C.. "Speaker Normalization Using Effiecient Frequency Warping Procedures," *Proc. ICASSP*, 353-356, 1996.
3. Pitz M., Ney H.. "Vocal Tract Normalization as Linear Transformation of MFCC," *Proc. EUROSPEECH*, 1445-1448, 2003.
4. Wang W., Zahorian S. A.. "Vocal Tract Normalization Based on Spectral Warping," *Proc. ICSLP*, 1185-1188, 2004.
5. Faria A., Gelbart D.. "Efficient Pitch-based Estimation of VTLN Warp Factors," *Proc. INTERSPEECH*, 213-216, 2005.
6. Laroche J., Dolson M.. " New Phase-Vocoder Techniques for Pitch-Shifting, Harmonizing and Other Exotic Effects," *Proc. 1999 IEEE Workshop on Applications of Signal Processing to Audio and Acoustic*, 91-94, 1999.
7. http://www.dspdimension.com/data/index.html
8. Gouva E. B., Stern R. M.. "Speaker Normalization through Formant-based Warping of The Frequency Scale," *Proc. EUROSPEECH*, 1139-1142, 1997.
9. Zhan P., Westphal M.. "Speaker Normalization Based on Frequency Warping," *Proc. ICASSP*, 1039-1042, 1997.
10. Magimai-Doss M., Stephenson T. A., Bourlard H.. "Using Pitch Frequency Information in Speech Recognition," *Proc. EUROSPEECH*, 2525-2528, 2003.
11. Zilca R. D., Kingsbury B., et al. "Pseudo Pitch Synchronous Analysis of Speech with Applications to Speaker Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):467-478, 2006.
12. Glavitsch U.. "Speaker normalization with respect to F_0: a perceptual approach," *TIK-Report Nr. 185*, 2005.
13. Liu J., Zheng T. F., et al. "Real-time Pitch Tracking Based on Combined SMDSF," *Proc. INTERSPEECH*, 301-304, 2005
14. Wang D., Zhu X.Y., Liu Y.. "Multi-Layer Channel Normalization for Frequency-Dynamic Feature Extraction," Journal of Software, v 12, n 9, September, 2005, p1523-1529
15. Young S., et al. *The HTK book (for HTK version 3.2).* Cambridge University Engineering Department, 2002

A Study of Knowledge-Based Features for Obstruent Detection and Classification in Continuous Mandarin Speech

Kuang-Ting Sung and Hsiao-Chuan Wang

Department of Electrical Engineering,
National Tsing Hua University, Hsinchu
g935925@oz.nthu.edu.tw, hcwang@ee.nthu.edu.tw

Abstract. A study on acoustic-phonetic features for the obstruent detection and classification based on the knowledge of Mandarin speech is proposed. Seneff auditory model is used as the front-end processor for extracting acoustic-phonetic features. These features are rich in their information content in a hierarchical decision process to detect and classify the Mandarin obstruents. The preliminary experiments showed that accuracy of obstruent detection is about 84%. An algorithm based on the information of feature distribution is applied to further classify the obstruents into stops, fricatives, and affricates. The average accuracy of obstruent classification is about 80%. The proposed approach based on the feature distribution is simple and effective. It could be a very promising method for improving the phone detection in continuous speech recognition.

Keywords: knowledge based approach, obstruent detection, obstruent classification.

1 Introduction

In typical Automatic Speech Recognition (ASR), a set of features is defined to specify the characteristics of speech in each frame. This set of features is used for recognizing all speech units. The statistical models based on this set of features are generated using speech databases. However, the corpus-based speech recognition approach cannot catch the specific characteristics of each individual phone, so that the performance of ASR is far from the performance of human speech recognition. Toward the next generation ASR, a paradigm integrating the knowledge sources with the recognition system was proposed [1][2]. This approach is based on the knowledge of articulatory phonetics and acoustic landmarks. Obstruents are the potential landmarks for ASR. Due to their noisy, dynamic, relatively short, weak, speaker- and context-dependent nature, the automatic detection and classification of obstruents are the most challenging tasks.

This study concerns the extraction of acoustic-phonetic features for the detection and classification of obstruents in the continuous Mandarin speech. Seneff auditory

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 95 – 105, 2006.

model is used as the front-end processor [3][4]. Some works based on auditory models had been reported [5][6][7][8][9]. Our focus is on obstruents of Mandarin speech. The proposed method is to combine the acoustic-phonetic knowledge in statistical frameworks. Then several sets of features derived from Seneff auditory model outputs are applied for the speech segmentation and the obstruent detection. Another set of features is used to further classify the obstruents into stops, fricatives, and affricates.

The remainder of this paper is organized as follows. The function of front-end processor is described in Section 2. The features for detecting silence, sonorant, and obstruent in the continuous Mandarin speech are derived in Section 3. The procedure of obstruent classification is presented in Section 4. The feature selection and statistical models for the obstruent classification are explained also. Section 5 shows the experiment results. A conclusion is given in Section 6.

2 Front-End Processor

The proposed system is composed of three subsystems: the front-end processor, the obstruent detector, and the obstruent classifier. The details of three subsystems are explained in the following subsections.

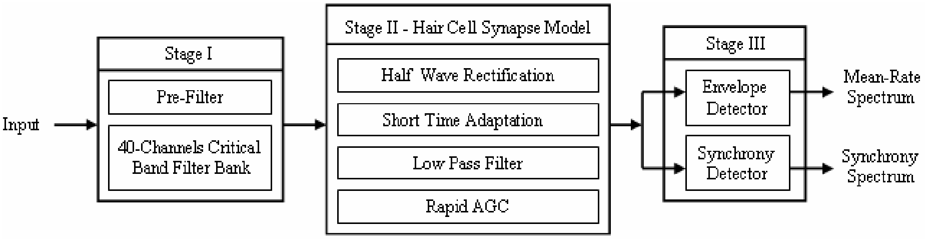


Fig. 1. System flow chart of Seneff auditory model

The front-end processor used in our system is a biologically-oriented filter-bank system. It is based on the system developed by Seneff [3][4]. The Seneff model is composed of 3 stages. The functional blocks are given in Fig.1. Stage I contains a pre-filter and a Bark-scaled filter bank. The former provides the function of eliminating high and low frequency components. The latter consists of 40 filters with 20dB/decade high frequency preemphasis. Stage II is called the hair cell synapse model. It consists of four steps to simulate the functions of transformation from basilar membrane vibration to probabilistic response and the properties of the auditory nerve fibers. Stage III gives two set of outputs, namely the Envelope Detector (ED) output and the synchrony detector (SD) output. Each output is a set of 40 components corresponding to 40 Bark-scale filters. The envelope detector (ED) output is the mean-rate output which enhanced sharpness of onset and offset of speech segments. The synchrony detector (SD) output can be modified to the Average Localized

Synchrony Detector (ALSD) output [5] which enhances spectral peaks due to vocal tract resonances. Our system is designed to use two kinds of outputs, ED and ALSD, for the extraction of acoustic-phonetic features. Fig. 2 demonstrates the example of ED output and ALSD output.

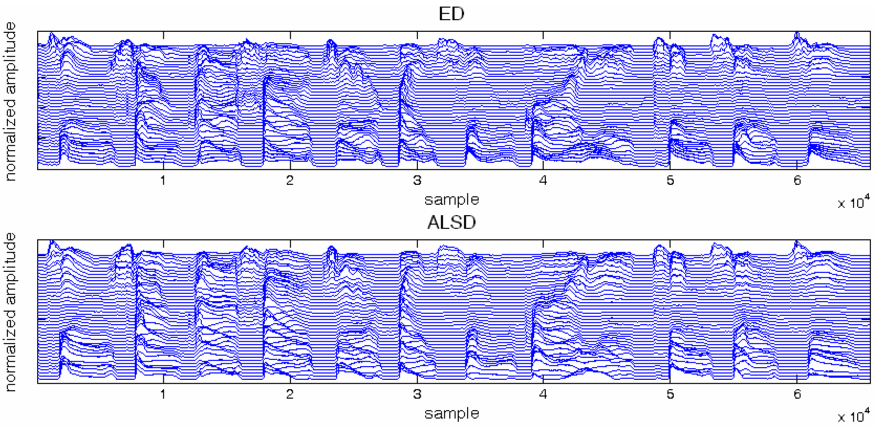


Fig. 2. An example of the outputs of ED and ALSD

3 Detection of Obstruents

To perform the obstruent detection, we categorize speech signal into three kinds of events, i.e., silences, sonorants, and obstruents. Fig. 3 shows the training phase and the testing phase of the obstruent detection. The 3-stage process starts with silence detection and follows by sonorant detection and obstruent detection. Finally, it uses continuity constraints to obtain the detection results.

In the following computation, the signal is sampled at 16 kHz. The frame length is 256 samples (16 ms) and the frames are overlapped by 84 samples (5.25 ms). Several

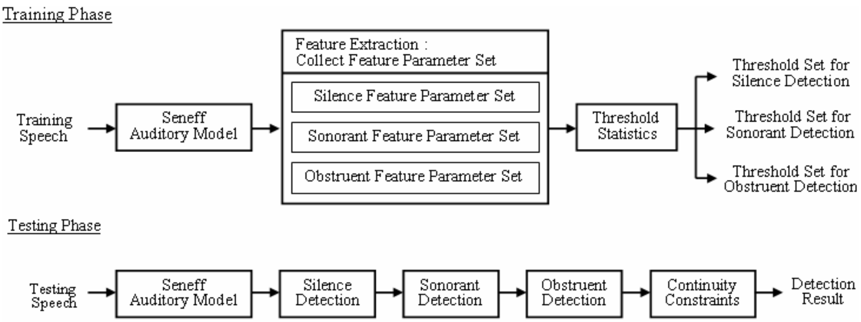

Training Phase
Training Speech
Seneff Auditory Model
Feature Extraction : Collect Feature Parameter Set
Silence Feature Parameter Set
Sonorant Feature Parameter Set
Obstruent Feature Parameter Set
Threshold Statistics
Threshold Set for Silence Detection
Threshold Set for Sonorant Detection
Threshold Set for Obstruent Detection
Testing Phase
Testing Speech
Seneff Auditory Model
Silence Detection
Sonorant Detection
Obstruent Detection
Continuity Constraints
Detection Result

Fig. 3. System flow chart of obstruent detection

features are examined based on the knowledge of articulatory phonetics. The histogram analysis is applied to determine which features will be used for categorizing the speech signal and for classifying obstuents. For obstruent detection, different feature detectors are used in three stages. The statistical information of each feature is obtained in the training phase by performing the histogram analysis of the extracted features.

3.1 Silence Detection

Three features are designed for the silence detection; the all-band normalized energy from ED ($ABNE_{ED,j}$), the all-band normalized energy from ALSD ($ABNE_{\mathrm{ALS}D,j}$), and the high-band normalized energy from ED ($HBNE_{ED,j}$). The subscript j is the frame index.

$$ABNE_{ED,j} = (\sum_{i=1}^{40} ED_{ij})/(\max_{j}\{\sum_{i=1}^{40} ED_{ij}\}) \tag{1}$$

$$ABNE_{ALSD,j} = (\sum_{i=1}^{40} ALSD_{ij})/(\max_{j}\{\sum_{i=1}^{40} ALSD_{ij}\}) \tag{2}$$

$$HBNE_{ED,j} = (\sum_{i=37}^{40} ED_{ij})/(\max_{j}\{\sum_{i=1}^{40} ED_{ij}\}) \tag{3}$$

The signal is sampled at 16 kHz. The output ED at j-th frame is a set of 40 components, ED_{ij} , i = 1, 2, …, 40, corresponding to the frequency range of 0-8 kHz. Similarly, the output ALSD at j-th frame is a set of 40 components, $ALSD_{ij}$, i = 1, 2, …, 40. The high bands of component index 37 to index 40 correspond to 3.5 kHz to 8 kHz. The distribution of each feature is obtained from the training data extracted from Mandarin speech database TCC300 which was recorded in the universities in Taiwan. Fig. 4 shows the histograms of three features. The threshold of equal error rate (EER) for each feature in discriminating the silence is marked also.

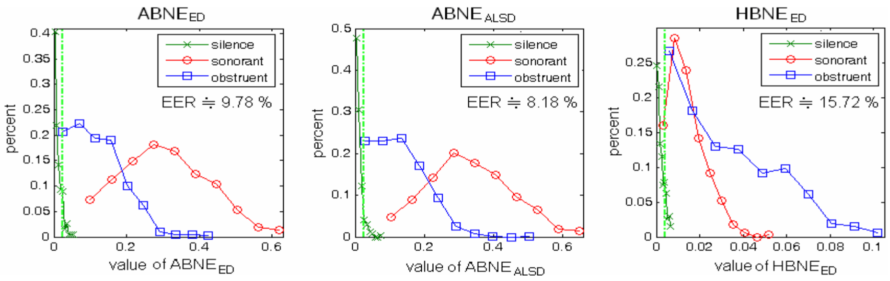


Fig. 4. Histograms of features used for silence detection

A frame is said to be a silence if either one of the following criteria is satisfied.

(1) Both $ABNE_{ED,j}$ and $ABNE_{ALSD,j}$ are less than their corresponding thresholds.

(2) $HBNE_{ED,j}$ is less than the threshold.

3.2 Sonorant Detection

Next step is to detect sonorants in the speech signal. Three features are designed for the sonorant detection; the low-band energy from ALSD ($LBE_{ALSD,j}$), the all-band energy from ALSD ($ABE_{ALSD,j}$), and the largest spectral peak location from ALSD ($LSPL_{ALSD,j}$).

$$LBE_{ALSD,j} = \sum_{i=1}^{8} ALSD_{ij} \tag{4}$$

$$ABE_{ALSD,j} = \sum_{i=1}^{40} ALSD_{ij} \tag{5}$$

$$LSPL_{ALSD,j} = \arg\max_{i} \{ ALSD_{ij} \} \tag{6}$$

Fig. 5 shows the histograms of three features. The threshold of equal error rate (EER) for for each feature in discriminating the sonorant is marked.

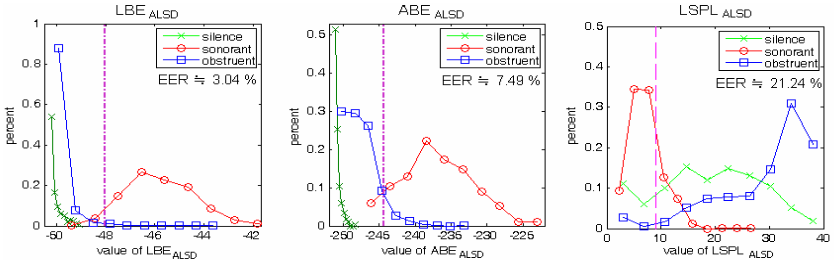


Fig. 5. Histograms of features used for sonorant detection

The low-band corresponds to the frequency range of 0-500 Hz. A frame is said to be a sonorant if either one of the following criteria is satisfied.

(1) Both $LBE_{ALSD,j}$ and $ABE_{ALSD,j}$ are greater than their corresponding thresholds.

(2) $LSPL_{ALSD,j}$ is less than the threshold.

3.3 Obstruent Detection

The last step is to detect obstruents in the region of no silence and no sonorant. Five features are designed for the obstruent detection; the low-band energy from ALSD

($LBE_{ALSD,j}$), the largest spectral peak location from ALSD ($LSPL_{ALSD,j}$), the spectral center of gravity from ED ($SCG_{ED,j}$), the energy difference between high-band and low-band from ED ($HLD_{ED,j}$), and the energy ratio of high-band to low-band from ED ($HLR_{ED,j}$). The first two features are defined in Eq (4) and Eq(6). The others are defined as follows;

$$SCG_{ED,j} = \frac{\sum_{i=1}^{40} i \times (ED_{ij} - \min_i\{ED_{ij}\})}{\sum_{i=1}^{40} (ED_{ij} - \min_i\{ED_{ij}\})} \qquad (7)$$

$$HLD_{ED,j} = (\sum_{i=31}^{40} ED_{ij}) - (\sum_{i=1}^{10} ED_{ij}) \qquad (8)$$

$$HLR_{ED,j} = (\sum_{i=31}^{40} ED_{ij}) / (\sum_{i=1}^{10} ED_{ij}) \qquad (9)$$

Fig. 6 shows the histograms of five features with the thresholds for discriminating the obstruent.

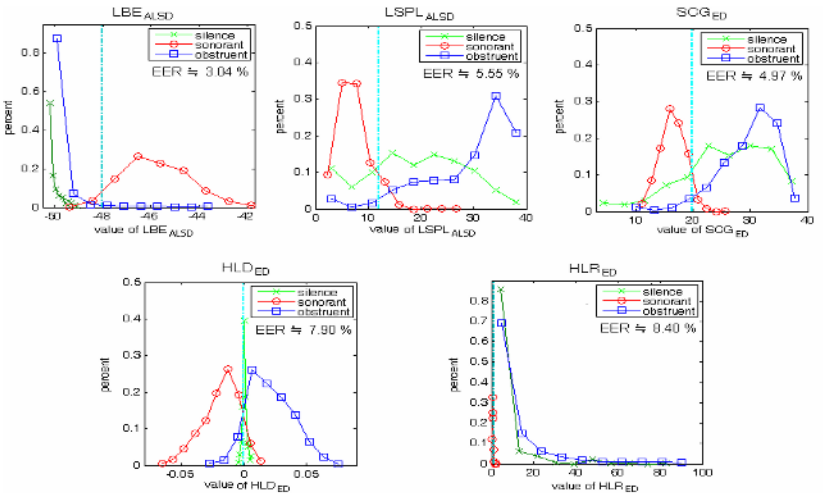


Fig. 6. Histograms of features used for obstruent detection

A frame is said to be an obstruent if either one of the following criteria is satisfied.

(1) $LBE_{ALSD,j}$ is less than the threshold

(2) Both $LSPL_{ALSD,j}$ and $SCG_{ED,j}$ are greater than their corresponding thresholds.

(3) $HLD_{ED,j}$ is greater than the threshold.

(4) $HLR_{ED,j}$ is greater than the threshold.

3.4 Post Processing

When a frame does not belong to any of three categories, an additional process is required to assign the undefined frame to a category of its closest neighbor frames. Other criteria for adjusting the detection result are based on the phonetic knowledge of Mandarin speech; (a) The duration of sonorant must be longer than three frame shifts (32 ms), (b) A segment must be ended with a sonorant, and (c) A single obstruent can not be a segment. An example of the obstruent detection is shown in Fig. 7. The utterance is a sentence spoken in Mandarin and labeled by Pinyin, “咳嗽藥常羼有其他成分(ke2 sou4 yao4 chang2 chan4 you3 qi2 ta1 cheng2 fen4).

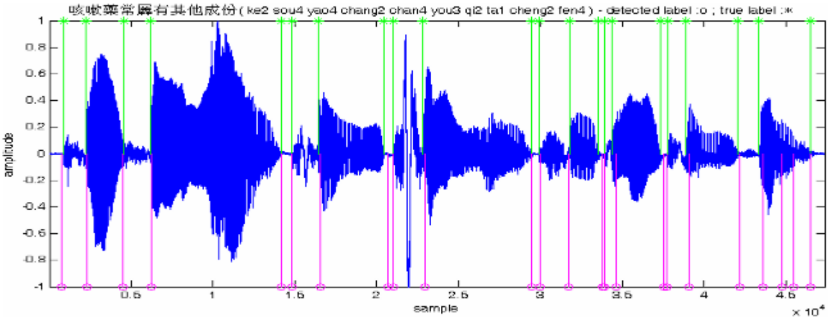


Fig. 7. An example of the obstruent detection results

The small circles (o) indicate the manually labeled boundaries. The stars (*) indicate the detected boundaries. Fig. 7 shows that most of obstruents are detected.

4 Classification of Obstruents

In Mandarin speech, the obstruents can be classified into fricatives, affricates, and stops. The flowchart for the obstruent classification is shown in Fig. 8. This process is performed on obstruent segments only. The scheme combines the acoustic-phonetic features and statistical model, GMM, to perform the automatical classification of obstruents. At first, the segment duration is used to discriminate stops and fricatives. The segment is a stop if its duration is less than 18 ms. The segment is a fricative if the duration is greater than 125 ms. For those obstruent segments with duration 18 – 125 ms, the Gaussian mixture model (GMM) method is applied to classify them into stops, fricatives, or affricates.

Six features are used in GMM classifier. They are the segment duration (DUR), the average zero crossing rate ($AZCR_j$), the spectral center of gravity from ED

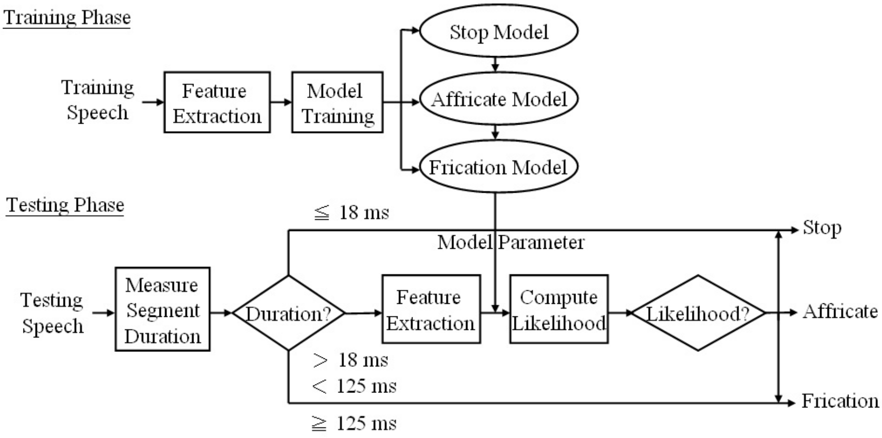


Fig. 8. System flow chart of obstruent classification

($SCG_{ED,j}$), the energy difference between high-band and low-band from ED ($HLD_{ED,j}$), the energy ratio of high-band to low-band from ED ($HLR_{ED,j}$), and the rate-of-rise-to-duration ratio (*RRDR*). The *RRDR* is computed by the following equations [6];

$$RRDR = \frac{\max_j\{RR_j\}}{DUR} \tag{10}$$

where

$$RR_j = RE_j + CCSD(SCG_j) \times \frac{N}{\max_j\{RE_j\}} \tag{11}$$

$$RE_j = CCSD(LBE_j) + CCSD(MBE_j) + CCSD(HBE_j) \tag{12}$$

$$LBE_{ED,j} = \sum_{i=1}^{13} ED_{ij} \tag{13}$$

$$MBE_{ED,j} = \sum_{i=10}^{27} ED_{ij} \tag{14}$$

$$HBE_{ED,j} = \sum_{i=37}^{40} ED_{ij} \tag{15}$$

$$CCSD(\bullet) = center\ clipping(smoothed\ difference(\bullet)) \tag{16}$$

Fig. 9 shows the histogram of six features.

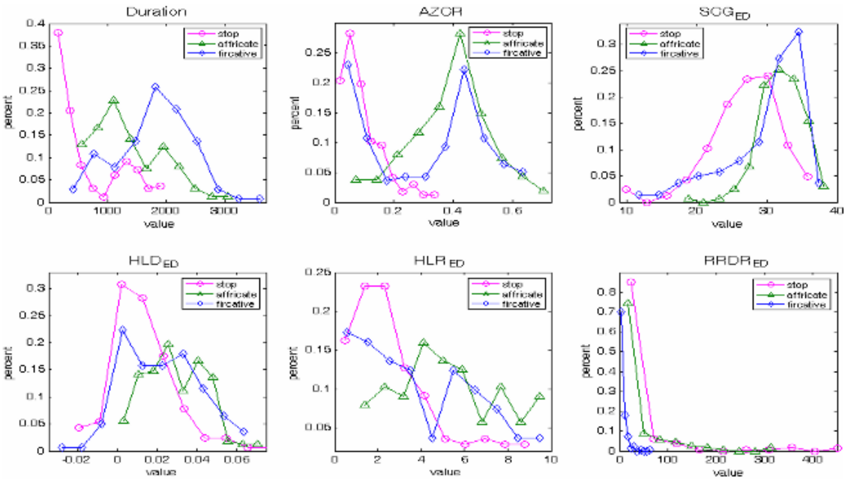


Fig. 9. Histograms of features used for obstruent classification

5 Experiments

The speech data sets for the experiments were extracted from Mandarin speech database, TCC300. Speech data of 2 male and 2 female speakers were selected as the training data. Each speaker provided 20 utterances. This training data set contains 268 silence segments, 521 sonorant segments, and 467 obstruent segments. Other speech data of 2 male and 2 female speakers, with 10 utterances per speaker, were selected as the testing data. The testing data set contains 124 silence segments, 254 sonorant segments, and 232 obstruent segments. All the selected data were labeled manually.

The obstruents in Mandarin speech is summarized in Table 1.

Table 1. Obstruents in Mandarin Speech (Not including the voiced consonants, such as nasals and laterals)

	Pinyin symbols
stops	/b/, /p/, /d/, /t/, /g/, /k/
affricates	/z/, /c/, /zh/, /ch/, /j/, /q/
fricatives	/f/, /s/, /sh/, /r/, /x/, /h/

For the obstruent detection test, the frame correct rates are given in Table 2. Table 3 shows the accuracy of obstruents classification.

Table 2. The frame correct rates (%)

	detected as silence	detected as sonorant	detected as obstruent
silence	76.3	12.2	11.5
sonorant	2.4	93.1	3.5
obstruent	2.7	12.9	84.4

Table 3. Accuracy of obstruent classification (%)

	detected as stop	Detected as affricate	detected as fricative
stop	92.2	5.2	2.6
affricate	8.6	74.3	17.1
fricative	10.9	19.6	69.5

From Table 2 we can find that the error rate of obstruent detection is about 15.6%. Table 3 shows that the average accuracy rate of obstruent classification is about 80%. Among the obstruents, stops get highest accuracy (92.2%).

6 Conclusions

This paper presents a preliminary study on the features for obstruent detection and classification in Mandarin Chinese. A method based on the combination of acoustic-phonetic knowledge and statistical models is proposed to detect silences, sonorants, and obstruents in the continuous Mandarin speech. The GMM method is applied to classify the obstruents into stops, affricates, and fricatives. The computation is simple and efficient. However, the accuracy rate of the proposed method is still low. To get more improvement, other feature selections and front-end processors need to be investigated.

Acknowledgments. This research was partially sponsored by the National Science Council, Taiwan, under contract number NSC94-2213-E007-021.

References

1. Chin-Hui Lee, "From Knowledge- Ignorant to Knowledge-Rich Modeling: A New Speech Research Paradigm for Next Generation Automatic Speech Recognition", in International Conference on Spoken Language Processing, ICSLP2004, Plenary Session , Jeju, Korea.
2. Kenneth N. Stevens, "Toward a model for lexical access base on acoustic landmarks and distinctive features", in J. Acoust. Soc. Am. 111 (4), pp. 1872-1891, April 2002.
3. Seneff, S., "A Joint Synchrony/ Mean Rate Model of Auditory Speech Processing", J. Phonetics, 16, pp. 55-76, 1988.
4. Seneff, S., "A Computational Model for the Peripheral Auditory System: Application to Speech Recognition Research", in IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 1983-1986, 1986.
5. Abdelatty Ali, A.M., " Auditory-Based Speech Processing Based on the Average Localized Synchrony Detection " , in IEEE International Conference on Acoustics, Speech, and Signal Processing, 2000.
6. Abdelatty Ali, A.M., "Segmentation and Categorization of Phonemes in Continuous Speech", Technical Report, TRCST25JUL98, Center for Sensor Technologies, University of Pennsylvania, 1998.
7. Aversano, G., "A New Text-Independent Method for Phoneme Segmentation", in IEEE International Conference on Circuit and System , 2001.

8. Hu Hongtao, "Temporal pre-classification for Chinese voiceless consonant speech", in IEEE International Conference on Signal Processing , 1996.
9. Abdelatty Ali, A.M., "An Acoustic-Phonetic Feature-Based System for the Automatic Recognition of Fricative Consonants", in IEEE International Conference on Acoustics, Speech, and Signal Processing, 1988.

Speaker-and-Environment Change Detection in Broadcast News Using Maximum Divergence Common Component GMM

Yih-Ru Wang

National Chiao Tung Univeristy,
1001 Ta Hseuh Road, Hsinchu
yrwang@cc.nctu.edu.tw

Abstract. In this paper, the supervised maximum-divergence common component GMM (MD-CCGMM) model was used to the speaker-and-environment change detection in broadcast news signal. In order to discriminate the speaker-and-environment change in broadcast news, the MD-CCGMM signal model will maximize the likelihood of CCGMM signal modeling and the divergence measure of different audio signal segments simultaneously. Performance of the MD-CCGMM model was examined using a four-hour TV broadcast news database. A result of 16.0% Equal Error Rate (EER) was achieved by using the divergence measure of CCGMM model. When using supervised MD-CCGMM model, 14.6% Equal Error Rate can be achieved.

Keywords: speaker-and-environment change detection, common component Gaussian mixture model, maximum divergence measure.

1 Introduction

The segmentation of audio signal is an important technology because large amounts of information were delivered through the audio signal such as broadcast news everyday. A good segmentation scheme is useful for further processing to categorize, archive and retrieve the information in broadcast news. Lots of studies had been done before in the audio signal segmentation problem and they can be categorized into two classes, one is feature-analysis based method and the other is metric-based method.

In feature-analysis based method, lots of distinctive features such as high ZCR ratio (HZCRR), low short time energy ratio (LSTER), spectrum flux (SF) [1], variance of the spectrum flux (VSF) and variance of the zero-crossing rate (VZCR) [2] were proposed in order to segmenting the complicate audio signal such as broadcast news.

The metric-based method wants to find a good measure which can indicate the statistics similarity/dissimilarity between two audio segments beside a candidate point. Many similarity measures of audio signals were proposed in the past, they included symmetric Kullback Leibler distance (KL2) [3], divergence shape distance [4] and Bayesian information criterion (BIC) [2, 5]. They are all derived from the Jeffrey divergence measure [6], in order to carry out a simple formula, the signal was

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 106 – 115, 2006.

assumed Gaussian distributed. In our previous study [7], a more precise signal modeling, the common component Gaussian mixture model (CCGMM) was used to model the statistics of audio signal segment. The Jeffrey divergence measure could be simplified into a discrete form in terms of the CCGMM coefficients. And, the performance of divergence measure using CCGMM coefficients was proved be better than the other metric-based methods.

In this paper, a supervised training algorithm which will maximize the divergence measure (MD) criterion of the CCGMM model was proposed. The maximum divergence CCGMM model can not only model the statistics of broadcast news audio signal but also can discriminate the signal of different speakers and environments by maximizing the divergence measure between them.

The paper is organized as follows. Section 2 describes the CCGMM-based divergence measure and the supervised training algorithm of maximum divergence measure CCGMM. Section 3 discusses the experimental results for the speaker-and-environment detection for a television broadcast news database. Some conclusions are given in the last section.

2 Modeling Audio Signal Using CCGMM and Maximum Divergence Training

The Jeffrey divergence measure [6] was used to measure the dissimilarity of two random variables based upon the information theory. It is derived from the average discriminating information between the two random signals. It can be expressed by

$$D(p_1,p_2)=\int\left[p_1(\mathbf{O})-p_2(\mathbf{O})\right]\ln\frac{p_1(\mathbf{O})}{p_2(\mathbf{O})}d\mathbf{O}\,. \tag{1}$$

where $p_1(\mathbf{O})$ and $p_2(\mathbf{O})$ are the probability distributions of the two signals which can be of two audio segments. In the our previous study, the distribution of two audio segments can be represented by the common component Gaussian mixture models (CCGMM), which was the mixture Gaussian density with common mixture components, i.e.,

$$p_n(\mathbf{O}|\lambda_n)=\sum_{i=0}^{M-1} c_{in}N(\mathbf{O}|\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i)\quad \forall n=1,2\,. \tag{2}$$

where $\lambda_n=\{(c_i,\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i);i=0,\cdots,M-1\}$ are the parameters sets of two signals.

Then, the divergence measure between two distributions, statistics of two adjacent audio segments, $\mathbf{O}^S,\mathbf{O}^{S'}$, can be approximated by [7]

$$D(\mathbf{O}^S,\mathbf{O}^{S'})\approx\sum_i(c_{is}-c_{is'})\ln\frac{c_{is}}{c_{is'}}\,. \tag{3}$$

Comparing the above divergence measure with the original definition of Jeffrey divergence shown in Eq. (1), we can find that they have the same form. And, the divergence measure of CCGMM, as shown in Eq. (3), can therefore be treated as a divergence measure of two discrete random variables and can be carried out easily. By using the CCGMM, the divergence of two complicated audio signal segments can be precisely evaluated.

In order to get better approximation in Eq. (3), a global diagonal covariance matrix $\mathbf{\Sigma}$ was used for all mixture components. The CCGMM for an audio segment becomes

$$p(\mathbf{O}^S \mid \lambda) = \sum_i p(\mathbf{O}^S, i \mid \lambda) = \sum_i c_{is} N(\mathbf{O}^S; \boldsymbol{\mu}_i, \mathbf{\Sigma}) . \tag{4}$$

The above CCGMM with global covariance matrix can be treated as using a set of Parzen windows with Gaussian kernels to estimate the distributions of signal sources. The mixture coefficients of CCGMM could, in fact, efficiently encode the data samples. And, the divergence of signal sources can be transformed into divergence of CCGMM coefficients.

The set of Gaussian kernels, $\{N(\mathbf{O} \mid \boldsymbol{\mu}_i, \mathbf{\Sigma}); i = 0, \cdots, M-1\}$, can be found from maximizing the likelihood of a training dataset, i.e.,

$$\underset{\lambda}{MAX} \prod_t \sum_i c_i N(\mathbf{O}_t \mid \boldsymbol{\mu}_i, \mathbf{\Sigma}), . \tag{5}$$

where O_t is the feature vector of the training signal at time *t*. In fact, the Gaussian kernel set is the universal background model (UBM) with global variance. For a segment of audio signal, $O_t^S, t \in S$, the CCGMM coefficients can be represented by, $\bar{c}_{is}; i = 0, \cdots, M-1$. And, the following re-estimation formula can be used to find the CCGMM coefficients of the audio segments.

$$\bar{c}_{is} = p(i \mid \mathbf{O}^S, \lambda) = \frac{1}{T} \sum_{t \in S} \frac{c_{is} N(\mathbf{O}_t^s; \boldsymbol{\mu}_i, \mathbf{\Sigma})}{\sum_j c_{js} N(\mathbf{O}_t^s; \boldsymbol{\mu}_j, \mathbf{\Sigma})}; i = 1, \cdots, K \ . \tag{6}$$

where *T* is the length of audio segment.

After the CCGMM coefficients of two adjacent audio segments were found, the divergence measure between them can be used to decide whether there is a signal change points between them. A simple threshold value for divergence measure was used in this paper to find the speaker-and-environment change points. A change point was detected if the local maximum in divergence curve was greater than the threshold, i.e., $D(O^{S_k}, O^{S_{k+1}}) > D_{TH}$, where S_k and S_{k+1} are two adjacent audio segments.

Although the CCGMM can precisely model the statistics of audio signal, but in the audio signal change point detection problem we want to maximize the divergence measure between two audio segments crossing a signal change point. In an audio signal contains a sequence of signal change points, denote as $\left\{T_k; k{=}1, \cdots, K\right\}$. In this paper, we want to find a CCGMM models, $\lambda = \{(c_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}); i = 0, \cdots, M-1\}$; which could simultaneously maximize the likelihood of all audio segments and also maximize the divergence measure between two audio segments crossing the change point, i.e.,

$$\underset{\lambda}{MAX} \prod_{S} \prod_{t \in S} p(\mathbf{O}_t^S | \lambda), \quad . \tag{7}$$

and

$$\underset{\lambda}{MAX} \sum_{k=1}^{K} D(\mathbf{O}^{S_k^-}, \mathbf{O}^{S_k^+}). \tag{8}$$

where S_k^-, S_k^+ are audio segments before and after signal change points T_k . Thus, the maximum divergence CCGMM (MD-CCGMM) model can use all the information provided by the training data. And the iterative supervised MD-CCGMM training algorithm was shown in following steps.

(1) First, a vector quantizer was used to get the initial model, $\tilde{\lambda} = \{(\tilde{c}_i, \tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}); i = 0, \cdots, M-1\}$. The Gaussian mixture mean, $\{\tilde{\boldsymbol{\mu}}_i; i = 0, \cdots, M-1\}$ in λ, were set to the codewords of the vector quantizer and the common covariance matrix, $\tilde{\boldsymbol{\Sigma}}$, was set to identity matrix.

(2) Find a new model $\lambda = \{(c_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}); i = 0, \cdots, M-1\}$ for all audio signal S which maximized the likelihood of CCGMM, i.e.,

$$\underset{\lambda}{MAX} \prod_{\forall S} \prod_{t \in S} \sum_i c_i N(O_t^S | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}) . \tag{7'}$$

And, the new model, λ, can be found by using the re-estimation algorithm of ordinary GMM.

(3) Given with $\lambda = \{(c_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}); i = 0, \cdots, M-1\}$ in step (2), we can find the mixture means, $\bar{\mu}_i$, of new CCGMM models $\bar{\lambda} = \{(c_{is}, \bar{\boldsymbol{\mu}}_i, \boldsymbol{\Sigma}); i = 0, \cdots, M-1\}$ which maximize the divergence measure of those audio segments before and after signal-change points, i.e.,

$$\underset{\lambda}{MAX} \frac{1}{K} \sum_k \sum_i (c_{is_k^-} - c_{is_k^+}) \log \left. \left(\frac{c_{is_k^-}}{c_{is_k^+}} \right) \right|_{\bar{\lambda}} . \tag{8'}$$

In order to get new mixture means, $\bar{\mu}_i$ in $\bar{\lambda}$, we can first express the $\{c_{is}; i=0,\cdots,M-1;\ S=(S_k^-, S_k^+ | \forall k)\}$ in Eq. (8') as

$$c_{is} = \frac{1}{T}\sum_{t\in S}\left[p(i|\mathbf{O}_t^S,\lambda)\right] = \frac{1}{T}\sum_t \frac{c_{is}N(\mathbf{O}_t^S;\boldsymbol{\mu}_i,\boldsymbol{\Sigma})}{\sum_j c_{is}N(\mathbf{O}_t^S;\boldsymbol{\mu}_j,\boldsymbol{\Sigma})}, \quad . \qquad (9)$$
$$\forall i = 0,\cdots,M-1, S = (S_k^-, S_k^+ \mid \forall k)$$

Then Eq. (8’) became a function of $\boldsymbol{\mu}_i$, but the new mixture means $\bar{\boldsymbol{\mu}}_i$ of MD-CCGMM model $\overline{\lambda_S}$ which maximize the divergence measure was still very complicate. In this paper, the steepest descent method was used to find $\bar{\boldsymbol{\mu}}_i$ which will increase the divergence measure of those audio segments before and after signal-change points.

We first find the derivation of CCGMM coefficients,

$$\frac{\partial}{\partial\boldsymbol{\mu}_k}\bar{c}_{is} = \frac{\partial}{\partial\boldsymbol{\mu}_i}\frac{1}{T}\sum_{t\in S}\frac{c_{is}N(\mathbf{O}_t^s;\boldsymbol{\mu}_i,\boldsymbol{\Sigma})}{\sum_j c_{js}N(\mathbf{O}_t^s;\boldsymbol{\mu}_j,\boldsymbol{\Sigma})}$$
$$= \begin{cases} \frac{1}{T}\sum_t\left[\left(p(k|\mathbf{O}_t^s,\lambda_s^{'})\left(1-p(k|\mathbf{O}_t^s,\lambda_s^{'})\right)\right)\Sigma^{-1}(\mathbf{O}_t^s-\boldsymbol{\mu}_k)\right] & i=k \\ \frac{1}{T}\sum_t\left[\left(-p(i|\mathbf{O}_t^s,\lambda_s^{'})\,p(k|\mathbf{O}_t^s,\lambda_s^{'})\right)\Sigma^{-1}(\mathbf{O}_t^s-\boldsymbol{\mu}_k)\right] & i\neq k \end{cases} \cdot \qquad (10)$$

Then, the gradient of divergence measure was shown as

$$\nabla\boldsymbol{\mu}_i = \frac{\partial}{\partial\boldsymbol{\mu}_i}\frac{1}{K}\sum_k\sum_j\left.\left(\sum_k(\bar{c}_{js_k^-}-\bar{c}_{js_k^+})\log\frac{\bar{c}_{js_k^-}}{\bar{c}_{js_k^+}}\right)\right|_{\lambda_s^{'}}$$
$$= \frac{\boldsymbol{\Sigma}^{-1}}{K}\cdot\sum_k\sum_j\left[\begin{array}{l}\left(\frac{\partial}{\partial\boldsymbol{\mu}_i}\bar{c}_{js_k^-}-\frac{\partial}{\partial\boldsymbol{\mu}_i}\bar{c}_{js_k^+}\right)\log\frac{\bar{c}_{js_k^-}}{\bar{c}_{js_k^+}} \\ +(\bar{c}_{js_k^-}-\bar{c}_{js_k^+})\left(\frac{1}{\bar{c}_{js_k^+}}\frac{\partial}{\partial\boldsymbol{\mu}_i}\bar{c}_{js_k^-}-\frac{1}{\bar{c}_{js_k^-}}\frac{\partial}{\partial\boldsymbol{\mu}_i}\bar{c}_{js_k^+}\right)\end{array}\right]. \qquad (11)$$

Finally, the $\bar{\boldsymbol{\mu}}_i$ can be updated using the following formula,

$$\bar{\boldsymbol{\mu}}_i = \boldsymbol{\mu}_i + \eta\nabla\boldsymbol{\mu}_i,\ i = 0,\cdots,M-1. \qquad (12)$$

where η is the step size of the steepest descent method. The above mixture mean re-estimation formula will move $\boldsymbol{\mu}_i$ toward the mean of the audio segment with larger weights and away from the mean of the audio segment with smaller weights.

(4) The steps (2) and (3) will be iterated in order to simultaneous maximize the likelihood of audio segments and divergence measure at signal change points.

3 Database and Experiments

3.1 Database

A television broadcast news database was used in the following experiments to evaluate the performance of the proposed method. It was recorded by the Public Television Service Foundation of Taiwan and is referred to as the Public Television Service News Database (MATBN) [8]. Each recording in the database consisted of a broadcast news episode of 60 minutes. A digital audio recorder (DAT) was used to record the database from the broadcasting machine. The signals were transformed to the form of 16-bit data with 16-kHz sampling rate. In the record, there included opening music, news report, weather report, and advertisement. And, the speakers included the studio anchors, field reporters, interviewees, weather anchors. The background conditions included clean, background music, noise and speech. The corpus was segmented, labeled and transcribed manually using the “Transcriber” developed by LDC. The transcripts were in BIG5-encoded form, with Standard Generalized Markup Language (SGML) tagging to annotate acoustics conditions, background conditions, story boundaries, speaker turn boundaries and audible acoustic events, such as hesitations, repetitions, vocal non-speech events, external noise, etc. Both orthographic transcription level and acoustic background level markers were extracted from the transcription information as correct answer of the following speaker-and-environment change detection experiments.

Table 1. Statistics of 4 hours PTSND database

Signal Conditions	Percent (in time)
Speech only	36.0%
Speech with background music	36.9%
advertisements	10.0%
Music or background sound only	6.5%
Silence	2.6%
Speaker types	Percent (in time)
Speech of anchors and weather reporters inside studio	25.2%
Speech of reporters, interviewees in the field	74.8%

One hour data in MATBN was used as the training data in our following experiment, and there were totally 407 speaker-and-environment change points in the training data. And, four hours data in MATBN, with 1797 change points, were used as the test data. Some statistics of environment conditions were shown in Table 1. From Table 1, we can find that there are many diverse audio sources and signal conditions in MATBN database that makes the signal-and-speaker change point detection more difficult. And, there are about 50 speakers - including anchor, weather reporter, reporters in the filed and interviewee were found in each recording. The speaker-and-environment conditions changed rapidly in MATBN, about 50% of them changed within 5 seconds.

3.2 Experiments and Results

In our experiments, all broadcast news recording were pre-emphasis by $1–0.97z^{-1}$ and segmented into 30-ms frames with 10-ms frame shift. Twelve mel-frequency cepstral coefficients (MFCCs) were then extracted for each frame and taken as feature vectors. We first used one hour recording data to train a GMM model and took all its mixture components as the common components of CCGMM and the other four hours recording data will be used as test database. And the window length used to find the CCGMM coefficients of audio signal is 3 sec (300 frames) and the number of mixtures used in CCGMM is 256. Then, divergence measures were computed for candidate points which were equally spaced every 0.5 second over the test database. In order to reduce the computation time, the CCGMM coefficients were computed for 0.5 second sub-windows. Then the CCGMM coefficient of 3 second analysis windows can be fond from the average of six sub-windows' CCGMM coefficients, as shown in Fig. 1. With the use of 3-second analysis window, a change point was considered missing if there were no change points detected within a 3-second window centered on the true change point in the following experiments.

First, only the maximum likelihood criterion was used to train the CCGMM and divergence measure was used to detect the speaker-and-environment change points. To show the effectiveness of the proposed CCGMM modeling method, an example is displayed in Fig. 2. In this example, the window length is 3 sec (300 frames) and the number of mixtures used in CCGMM is 256. As shown in Fig. 2(a), there are 6 transcription level changes and 3 background condition changes in 50 sec audio signal. In Fig. 2(c), CCGMM weights of four pairs of consecutive windows are displayed. Weights of 10 common components corresponding to the largest weights of the second window are shown in gray level. It can be found from the figure that weights of the second and fourth window-pairs, which correspond to change points, are very different to each other.

The false alarm rate (FAR) vs. miss detection rate (MDR) curves of the test data, with different threshold values for divergence measure, were shown in Fig. 3. And, a result of 16.6% equal error rate (EER) can be achieved.

Then, the training algorithm described in section 3 was used to train a new supervised maximum divergence CCGMM (MD-CCGMM) model. By using the

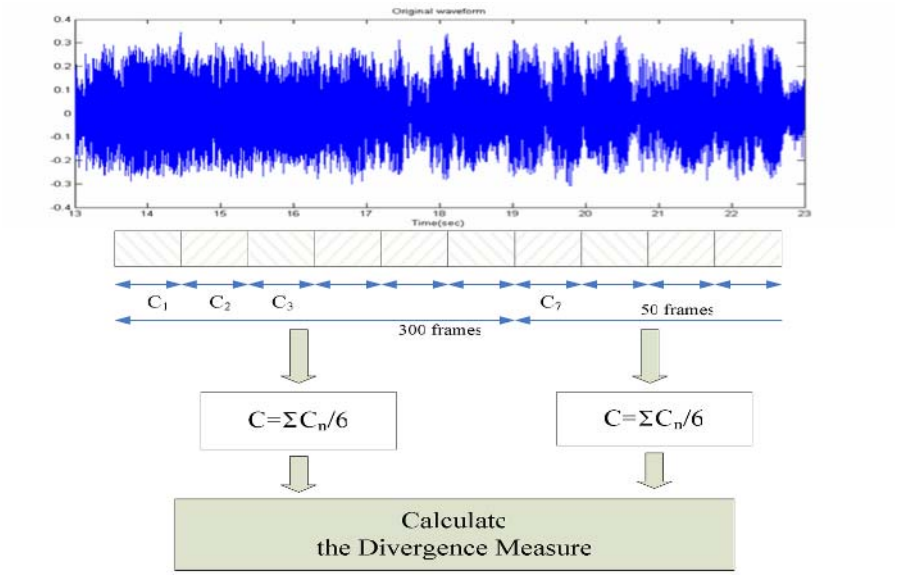


Fig. 1. The block diagram of CCGMM coefficients extraction

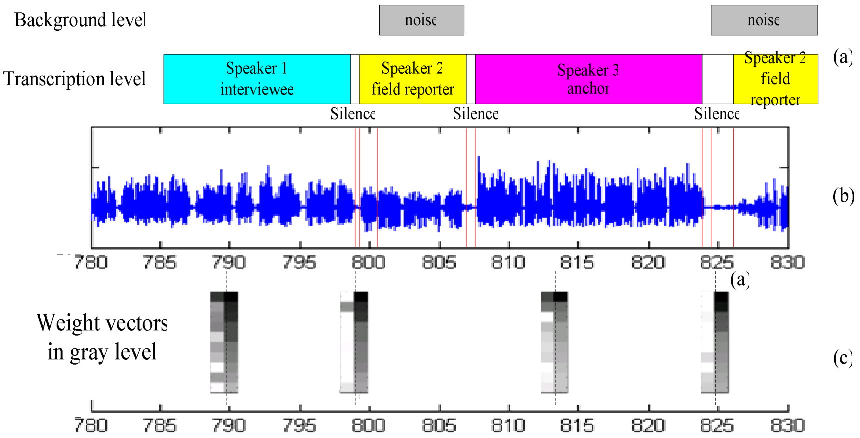


Fig. 2. An example of the proposed CCGMM-based method of speaker-and-environment change detection in broadcast news signals: (a) transcription information, (b) waveform (vertical lines indicate marks of change points), (c) the largest 10 CCGMM weights of for window pairs

MD-CCGMM training algorithm, the average divergence measure at speaker-and-environment change points for training data was increased from 56.38 to 59.36 and the likelihood of CCGMM will slightly decrease. But, the EER of testing data only decreased to 16.5%.

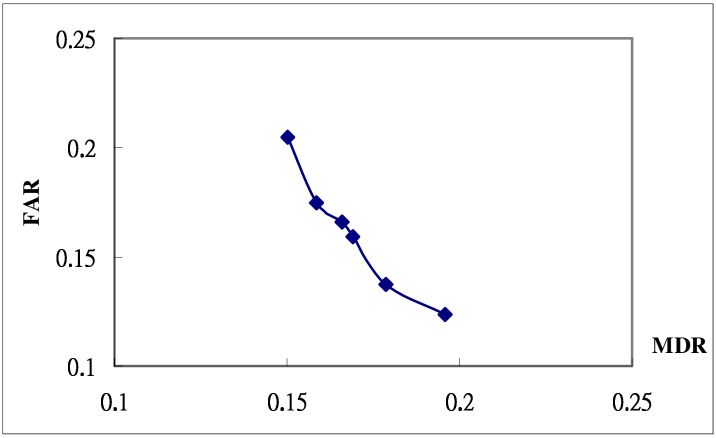


Fig. 3. The FAR-MDR curves of the CCGMM speaker-and-environment change detection scheme

Because the MD-CCGMM is a supervised learning algorithm, compare with CCGMM, more training data were needed. So, in the following experiment, two hours recordings were used for training and the other three hours recordings were used for testing. There are totally four different anchors in five recordings. One recording with female anchor and one with male anchor were choice as the training data. Now, there are totally 816, 1388 change points in training and test data. And, the false alarm rate vs. miss detection rate curves of the test data with different threshold values of signal divergence measure were also shown in Fig. 4. A result of 16.0% EER was achieved by using the CCGMM model, the performance will be slightly better when more data was used to train the CCGMM. When using the supervised MD-CCGMM model, EER reduce to 14.5%. About 10% EER reduction can be achieved by using the proposed supervised MD-CCGMM algorithm.

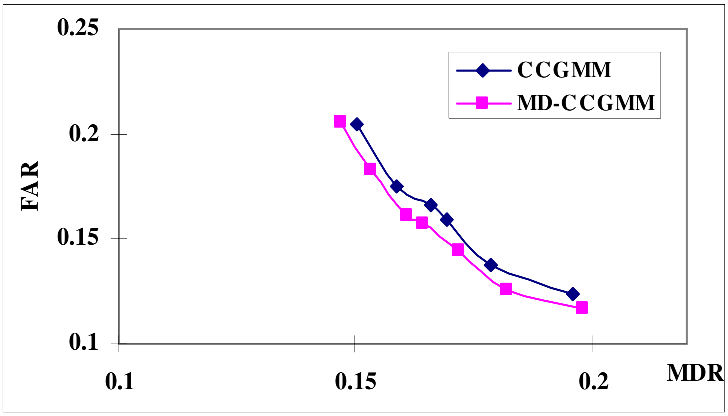


Fig. 4. The FAR-MDR curves of the CCGMM and MD-CCGMM speaker-and-environment change detection scheme

4 Conclusions

In this paper, the supervised MD-CCGMM model was used in broadcast news change-point detection problem. The MD-CCGMM model can simultaneously maximize the likelihood of signal model and divergence measure between signal change points was introduced. The supervised MD-CCGMM can get better performance than the unsupervised CCGMM method, but more training data were needed.

Acknowledgments. This work was supported by the National Science Council, Taiwan, ROC, under the project with contract NSC 94-2218-E-009 -007, 94-2213-E-009-134.

References

1. E. Scheirer and M. Slaney : Construction and Evaluation of a Robust Multifeature Music/Speech Discriminator, Proc. ICASSP 97, vol II, pp 1331-1334. IEEE, April 1997.
2. P. Angkititrakul, J. H. L. Hansen, S. Baghaii : Cluster-dependent Modeling and Confidence Measure Processing for In-Set/Out-of-Set Speaker Identification, Interspeech-2004/ICSLP-2004, pp. ThC1604 p.15(1-4), Oct. 2004.
3. M.A. Siegler, U. Jain, B. Raj, R. M. Stern : Automatic Segmentation, Classification of Broadcast News Audio, *Proc. DARPA speech recognition workshop*, pp. 97-99, 1997.
4. L. Lu, H.-J. Zhang : Speaker change detection and tracking in real-time news broadcasting Analysis, *Proc. of ACM Multimedia 2002*, pp. 602-610.
5. Scott Shaobing Chen, P.S. Gopalakrishnan : Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion, *Proc. DARPA speech recognition workshop*, 1998.
6. H. Jeffreys : An Invariant Form for the Prior Probability in Estimation Problems, *Proc. Roy. Soc. Lon., Ser. A*, no. 186, 453-461, 1946.
7. Yih-Ru Wang and Chi-Han Huang : Speaker-and-environment Change Detection in Broadcast News using the Common Component GMM-based Divergence Measure , Proc. of ICSLP 2004, Jeju island, , pp. 1069-1072, Oct. 2004.

UBM Based Speaker Segmentation and Clustering for 2-Speaker Detection

Jing Deng, Thomas Fang Zheng, and Wenhu Wu

Center for Speech Technology, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing, 100084
dengj02@mails.tsinghua.edu.cn, fzheng@tsinghua.edu.cn,
wuwh@tsinghua.edu.cn

Abstract. In this paper, a speaker segmentation method based on log-likelihood ratio score (LLRS) over universal background model (UBM) and a speaker clustering method based on difference of log-likelihood scores between two speaker models are proposed. During the segmentation process, the LLRS between two adjacent speech segments over UBM is used as a distance measure, while during the clustering process, the difference of log-likelihood scores between two speaker models is used as a speaker classification criterion. A complete system for NIST 2002 2-speaker task is presented using the methods mentioned above. Experimental results on NIST 2002 Switchboard Cellular speaker segmentation corpus, 1-speaker evaluation corpus and 2-speaker evaluation corpus show the potentiality of the proposed algorithms.

Keywords: Speaker segmentation, Speaker clustering, Multi-speaker, Speaker Detection.

1 Introduction

In real-world speaker verification tasks over telephone, there is an increasing demand that speaker verification systems can verify one specific speaker whether in a conversation or not. One of the solutions to this demand is speaker segmentation and clustering. The aim of speaker segmentation and clustering is to segment an *N*-speakers' conversation into speech segments containing the voice of only one speaker (segmentation process) and to merge those speech segments belonging to a same speaker into one speech segment (clustering process). After speaker segmentation and clustering, a multi-speaker verification task can be simplified into several *N* single-speaker verification tasks. Generally, no *a priori* information is available on the number and identity of speakers involved in the conversation.

Previous researches have focused on two directions, distance based and model based. The former does not require any *a priori* information, but it is difficult to accurately describe the characteristics of a speaker with short speech segments which often occur in conversations over telephone and hence will result in a dissatisfactory performance during the clustering process. Methods in this direction include Bayesian Information Criterion (BIC) [1], [2], [3], Generalized Likelihood Ratio (GLR) [4], [5], Kullback-Leibler (KL) Distance [6], [7], DISTBIC [8], *etc*. The latter can achieve

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 116 – 125, 2006.

a satisfactory result by building a model for each speaker in the audio recording and then using a global maximum likelihood score to find the best time-aligned speaker sequence (usually by using Viterbi algorithm). One of the difficulties in model based method is how to accurately build initial speaker models. The model based systems include LIA [9], ELISA [10], [11], *etc*.

Usually, there are many short speech segments in conversations over telephone. Distance based segmentation criteria, such as BIC, have some difficulties in dealing with them [8]. The reason is that it is difficult to estimate the characteristics of a speaker with short speech segment. Model based segmentation can well deal with this issue, however, they need *a priori* knowledge of speakers in the conversation. In order to well describe the characteristics of short speech segments, in this paper, UBM is used as *a priori* knowledge of speakers during segmentation process. Given two adjacent short speech segments belonging to a same speaker, the log-likelihood ratio score (LLRS) of them over UBM is small, and vice versa. So LLRS over UBM is used as a distance measure for speaker segmentation.

After segmentation, a conversation is divided into several speech segments. But the identity of each speech segment and the number of speakers are unknown. Because most conversations over telephone each contain only two speakers, the number of speakers in a conversation is set to 2 in this paper. Conventional speaker clustering methods mainly focus on finding out the closest speech segments while in this paper a method based on the difference of log-likelihood score between two speaker models is proposed to identify one speech segment as speaker *A* if it is the farthest one from speaker *B*. Over the NIST 2002 2-speaker segmentation Switchboard set, a system integrated with the proposed method can achieve a frame error rate of 6.8%, which will be detailed later.

This paper is organized as follows. The speaker segmentation based on LLRS will be presented in Section 2, and the proposed speaker clustering method will be described in Section 3. In Section 4, experiments and results will be described. Finally, conclusions and perspectives will be given in Section 5.

2 Speaker Segmentation Based on LLRS over UBM

In this paper, a simple segmentation criterion based on LLRS over UBM is used. First, acoustic features are extracted from the input speech. Then the acoustic features are divided into several decision windows by a sliding window with a 2-second width and a 0.1-second shift. In each decision window, the acoustic features are divided into two parts $X_1=(x_1, x_2, ..., x_i)$ and $X_2=(x_{i+1}, x_{i+2}, ..., x_N)$; and $LLRS\ (i)$ between them is defined as

$$LLRS(i) = abs\left(L\left(X_1 \mid UBM\right) - L\left(X_2 \mid UBM\right)\right) \tag{1}$$

where i was set to the half position of the decision window. Because there may be some silence or noise in one decision window, the log-likelihood score of a speech frame over UBM is used as a measure to decide whether current frame is a speech frame or a non-speech frame. The bigger the log-likelihood score, the more likely current frame is a speech frame. A similar process is proposed in [12] which used the log-likelihood score of one speech segment over UBM to separate the speech segment into three groups: *confidential* speech frames, *doubtable* speech frames, and *non*-speech frames. So in Equation (1), the acoustic features in each half decision window used to calculate the log-likelihood score are those whose scores are among the top half.

Finally, we can get a sequence of LLRS and the standard deviation σ can be estimated accordingly. In the LLRS plot (showed in Fig. 1), a peak is assumed to be a possible speaker turn point if

$$\left|max-min_l\right|>\alpha\sigma \quad \text{and} \quad \left|max-min_r\right|>\alpha\sigma \tag{2}$$

where α is an experiential value which is set to 0.5 in experiments in this paper, *max* is the LLRS at the peak position, and min_l and min_r are the left and right minima around the peak value point, respectively. More details about Equation (2) were described in [8].

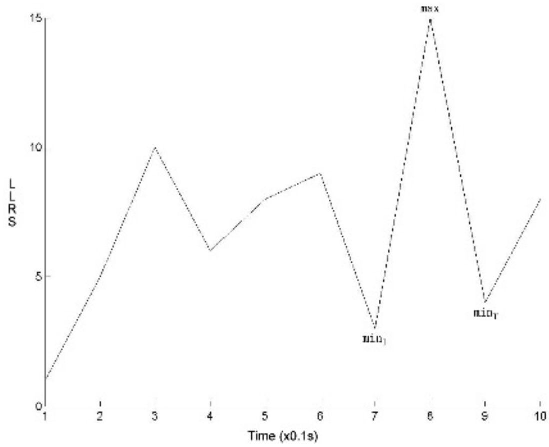


Fig. 1. LLRS plot: decision of a speaker turn

3 Speaker Clustering Based on Difference of Log-Likelihood Score Between Two Speaker Models

The goal described here in this section is to cluster speech segments with a same speaker identity. As mentioned above, the number of speakers in one conversation is 2. So given two speaker models (*A* and *B*) and several speech segments $\{X_i, i=1, 2, \ldots, N\}$, speech segment X_j is regarded to most likely belong to speaker model *A* if

$$j=\arg\max_i\left(L\left(X_i \mid A\right)-L\left(X_i \mid B\right)\right) \tag{3}$$

where *L*(.) is the log-likelihood function. After speaker segmentation, there are many short speech segments which are not long enough to well train a speaker model. In order to solve this problem, a multi-stage clustering strategy is used. First, a UBM with a small number of components is used to select suitable speech segments for initial model training. Then with sufficiently long speech segments, speaker models can be well trained from a UBM with large number of components. The proposed speaker clustering method is described as follows.

Stage 1. Initial clustering

1.1 First an initial speaker model S_0 is adapted on the whole test utterance from *UBM*1 by MAP with only mean vector changed.

1.2 After speaker segmentation, all the speech segments are scored on S_0. The speech segment with the maximal log-likelihood score and longer than 2 seconds is selected for use of adapting speaker model S_1 from *UBM*1.

1.3 The remained speech segments are scored against S_0 and S_1, respectively. The difference of log-likelihood score, ΔS, is defined as

$$\Delta S = L(X \mid S_0) - L(X \mid S_1) \tag{4}$$

where X is the acoustic feature sequence from a speech segment. The bigger the ΔS is, the more likely X not belongs to S_1. The speech segment with the maximal ΔS and longer than 2 seconds is selected for use of adapting speaker model S_2 from *UBM*1.

1.4 Score the remained speech segments against S_1 and S_2. From those speech segments with score $L(X|S_1)$ bigger than $L(X|S_2)$, the speech segment with the maximal ΔS_{12} and longer than 1 second is selected for use of updating S_1, where $\Delta S_{12}= L(X|S_1)-L(X|S_2)$. From those speech segments with score $L(X|S_2)$ bigger than $L(X|S_1)$, the speech segment with the maximal ΔS_{21} and longer than 1 second is selected for use of updating S_2, where $\Delta S_{21}= L(X|S_2)-L(X|S_1)$.

1.5 Repeat 1.4 until there is no speech segment longer than 1 second.

1.6 Use S_1 and S_2 to calculate ΔS_{12} in speech segments belonging to S_1 and ΔS_{21} in speech segments belonging to S_2.

Stage 2. Refine the clustering

2.1 Adapting a new speaker model S_1 from *UBM*2 with speech segments belonging to previous S_1 which ΔS_{12} is among the top half.

2.2 Adapting a new speaker model S_2 from *UBM*2 with speech segments belonging to previous S_2 which ΔS_{21} is among the top half.

2.3 Score each speech segment against S_1 and S_2, respectively. If ΔS_{12} is positive, the speech segment is assigned to S_1, otherwise to S_2. Meanwhile, calculate ΔS_{12} on those speech segments belonging to S_1 and ΔS_{21} on speech segments belonging to S_2 for use in stage 3.

Here, *UBM*1 and *UBM*2 can be of different component sizes. In our experiments, *UBM*1 contains 16 components and *UBM*2 contains 1,024 components.

4 Experiments and Results

The features were extracted from speech signal at a frame size of 20 milliseconds every 10 milliseconds. The pre-emphasis factor was set to 0.97. The Hamming windowing was applied to each pre-emphasized frame. After that, a 256-point FFT was calculated for each frame and a bank of 30 triangular Mel filters were used.

Finally DCT was performed and 16-dimensional MFCC coefficients with the delta coefficients were obtained for each frame.

The baseline system in our experiments was based on the Gaussian Mixture Model-Universal Background Model (GMM-UBM) [13] with the UBM gender-independent, tree-structured [14], and containing 1,024 mixtures. No score normalization method was performed.

4.1 Segmentation Results

We tested the segmentation and the clustering methods on the NIST 2002 Switchboard Cellular speaker segmentation corpus. This corpus contains 199 test segments (two minutes each) involving only two speakers (at an 8 kHz sampling rate). The evaluation method was the NIST official scoring (version 07) [15] which is a frame based error rate protocol. Table 1 shows the accuracy of initial speech segments selection for model S_1 and S_2 in the clustering process (Steps 1.1 to 1.3). The segmentation results of LIA and the proposed method on NIST 2002 Switchboard Cellular speaker segmentation corpus are showed in Table 2.

The LIA system is an HMM based speaker segmentation system. Each state of the HMM characterizes a speaker and the transitions model the changes between speakers. During the segmentation, the HMM is generated using an iterative process, which detects and adds a new state (i.e. a new speaker) at each iteration.

We also compared the false alarm rates and the miss detection rates among BIC, GLR, DISTBIC, and the proposed method, listed in Table 3.

Table 1. Initial speech segments selection results on NIST 2002 Switchboard Cellular speaker segmentation corpus

Error Type	Error Time Rate (%)
Missed Speaker Time	0.1
False Alarm Speaker Time	0.3
Speaker Error Time	0.4

Table 2. NIST 2002 speaker segmentation results for Switchboard Cellular speaker segmentation corpus

System	Missed Speaker Time	False alarm Speaker Time	Speaker Error Time
LIA [16]	0.0%	0.0%	7.4%
Propose method	0.1%	0.1%	6.6%

Table 3. Segmentation performance comparison of BIC, GLR, DISTBIC, and the proposed method on the NIST 2002 Switchboard Cellular Speaker Segmentation Corpus

System	FAR(%)	MDR(%)
BIC	25.2	35.6
GLR	33.2	19.5
DISTBIC	30.8	20.3
Propose method	29.3	18.9

The total segmentation error rate of CLIPS is 8.6% and the fusion of LIA and CLIPS can achieve an error rate of 5.7% on NIST 2002 Switchboard Cellular speaker segmentation corpus [10]. Compared with LIA and CLIPS, the proposed method can achieve a comparative performance.

4.2 1-Speaker Detection (1D) Results

The training set contains 330 speech segments (two minutes each) by 139 males and 191 females. The test set contains 3,570 speech segments by 1,442 males and 2,128 females with about 15 to 45 seconds for each segment. The detection results are given in Fig. 2. Comparison result of LIA is showed in Fig 3 [16].

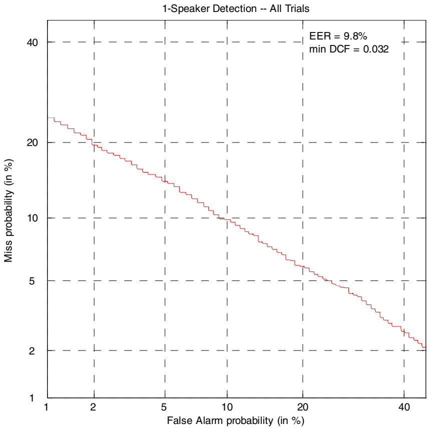


Fig. 2. 1-speaker detection results, NIST 2002 evaluation

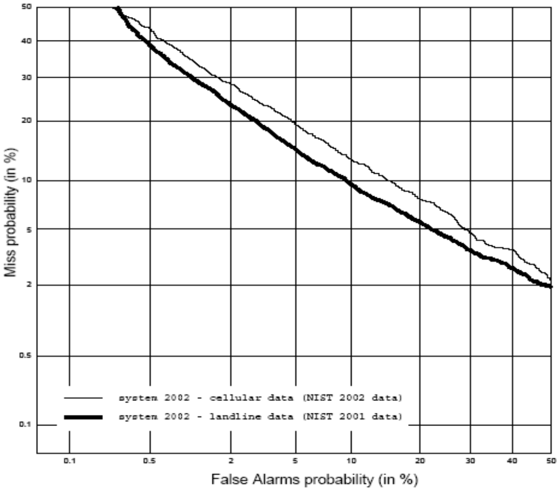


Fig. 3. LIA 1-speaker results on NIST 2002 cellular data and NIST 2001 landline data

4.3 1-Speaker Training, 2-Speaker Detection (1T- 2D) Results

Here, 1T-2D means using 1-Speaker speech segment for training and using 2-Speaker speech segment for detection. The training set here was same as that used in 1D evaluation. The test set contains 1,470 speech segments (one minute each) by 2 speakers (two males, two females or one male - one female). The detection results are given in Fig. 4.

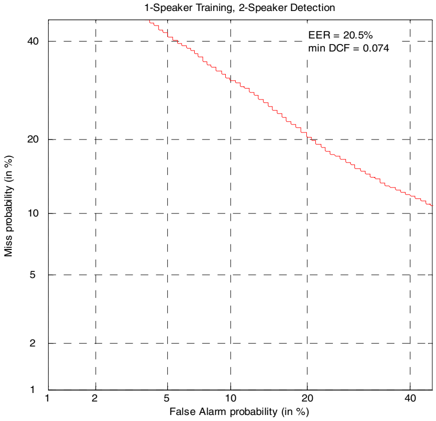


Fig. 4. 1T-2D results, NIST 2002 evaluation

4.4 2-Speaker Detection (2D) Results

This evaluation illustrates the effect of training a target speaker model from three 2-speaker audio files. No *a priori* information was provided except that the target speaker was the only speaker in each of the three files. The training set contains 309 target speakers (131 males and 178 females) and the test set contains 1,460 segments, each with an average duration of one minute spoken by two speakers. The training process is illustrated in Fig.5.

For each 2-speaker audio file, two final speech segments will be obtained by using the proposed segmentation and clustering methods. For each final speech segment, a speaker model can be trained form UBM with mean vectors changed only. That is to say, given three 2-speaker audio files, six speaker models can be obtained finally. Let S_1 and S_2 be any two speaker models form two audio files respectively, where the i-th components in S_1 and S_2 are defined as $\left(w_i, \mu_i^1, \Sigma_i\right)$ and $\left(w_i, \mu_i^2, \Sigma_i\right)$, respectively. The KL distance between S_1 and S_2 was calculated as

$$KL(S_1, S_2) = \sum_{m=1}^{M}\left(w_i \cdot \left(\mu_i^2 - \mu_i^1\right)^T \cdot \left(\mu_i^2 - \mu_i^1\right) \cdot \Sigma_i^{-1}\right) \tag{5}$$

where M is the number of components in each model.

As showed in Fig. 5, if the KL distance between X_1 and Y_1 is smaller than that between X_1 and Y_2, X_2 and Y_1, or X_2 and Y_2, speech segments T_1 and T_3 will be merged

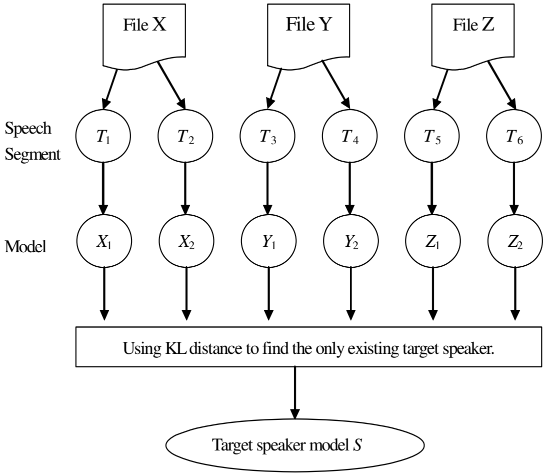


Fig. 5. Multi-speaker training process

together. Finally, a target speaker model S can be obtained from these three 2-spekaer audio files.

The detection results are given in Fig. 6. Comparison result of LIA is showed in Fig. 7 [16].

4.5 Discussion

It can be seen that there exist two large losses: one lies in the performance between 1D and 1T-2D, the other lies in the performance between 2D and 1T-2D. The loss comes from several aspects: (1) there existed many short speech segments and noisy speech segments that might cause errors in segmentation and clustering; (2) there

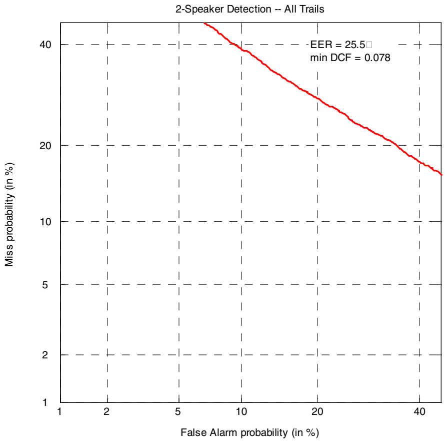


Fig. 6. 2-speaker detection, NIST 2002 evaluation

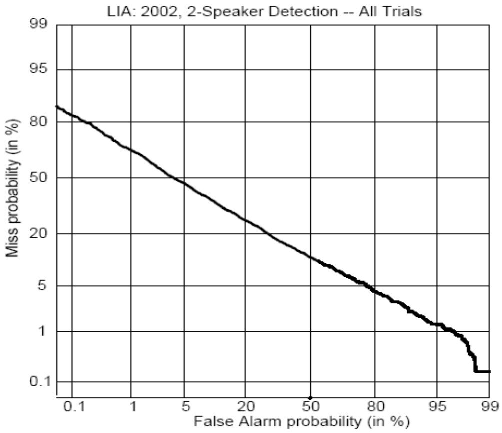


Fig. 7. LIA 2-speaekr result, NIST 2002 evaluation

existed many speech segments spoken by two speakers simultaneously; (3) there existed some mistakes in the multi-speaker training process which might lead to a bad target model; (4) the average duration of speech segments used in 1D was longer than that used in the other two detections; and (5) the errors caused by speaker segmentation can not be corrected by the clustering process.

5 Conclusions

In this paper, a speaker segmentation method based on LLRS over UBM and a speaker clustering method based on difference of log-likelihood scores between two speaker models are proposed. And a complete system with related experiments and results for NIST 2002 two-speaker task is presented. The target models are trained from several multi-speaker speech segments and the tests are also done with 2-speaker files.

The proposed speaker segmentation and clustering methods can achieve a frame error rate of 6.8% on NIST 2002 Switchboard Cellular speaker segmentation corpus. And for 1T-2D, the system achieves an EER of 20.5%, and for 2-speaker detection, the system achieves an EER of 25.5%. The performances of the proposed method on NIST 2002 Switchboard Cellular speaker segmentation corpus, the 1D and 2D tasks are close to that of LIA [16].

Though the segment result seems accurate enough for the task, the performances of 1T-2D and 2D are less satisfactory. Something must be done in order to decrease the detection errors: (1) perform re-segmentation with the speaker models trained in clustering phase; (2) discard the speech segments with bad Signal-to-Noise Ratio (SNR) or overlapped by several speakers; (3) improve the matching strategy during multi-speaker training in order to obtain a more accurate target speaker model.

References

1. Rissanen, J. Stochastic Complexity in Statistical Inquiry. Series in Computer Science, 1989, Vol. 15. World Scientific, Singapore, Chapter 3
2. Chen, S.S., Gopalakrishnan, P.S. Speaker environment and channel change detection and clustering via the Bayesian Information Criterion. In: DARPA Speech Recognition Workshop, 1998
3. Rissanen, J. Stochastic Complexity in Statistical Inquiry. Series in Computer Science, 1989, Vol. 15. World Scientific, Singapore, Chapter 3
4. Gish, H., Siu, M.-H., Rohlicek, R. Segregation of speakers for speech recognition and speaker identification. In: IEEE International Conference on Acoustics Speech and Signal Processing, 1991. 873-876
5. H. Gish and M. Schmidt. Text-independent speaker identification. IEEE Signal Processing Mag. 1994, 11:18-32
6. Siegler, M.A., Jain, U., Raj, B., Stern, R.M. Automatic segmentation classi®cation and clustering of broadcast news audio. In: DARPA Speech Recognition Workshop, 1997. 97-99
7. J. P. Campbell, Jr. Speaker recognition: A tutorial. Proc. IEEE, 1997. 9(85):1437 – 1462
8. P. Delacourt , CJ Wellekens. DISTBIC: a speaker-based segmentation for audio data indexing, Speech Communication, Sept. 2000, (32):111-126
9. Sylvain Meignier, Jean-FranÇois Bonastre, and Stéphane Igounet. E-HMM approach for learning and adapting sound models for speaker indexing. In 2001: A Speaker Odyssey, Chania, Crete, June 2001. 175-180
10. D. Moraru, S. Meignier, L. Besacier, J.-F. Bonastre, Y. Magrin-Chagnolleau. The ELISA consortium approaches in speaker segmentation during the NIST2002 speaker recognition evaluation, In Proceedings of International Conference on Acoustics Speech and Signal Processing (ICASSP 2003), Hong Kong, 2003. (2): 89 – 92
11. D. Moraru, S. Meignier, C. Fredouille, L. Besacier, J.-F. Bonastre. The ELISA consortium approaches in broadcast news speaker segmentation during the NIST2003 rich transcription evaluation, In Proceedings of International Conference on Acoustics Speech and Signal Processing (ICASSP 2004), Montreal, Canada, 2004
12. T. Wu, L. Lu, K. Chen, and H. Zhang. UBM-based real-time speaker segmentation for broadcasting news. In Proc. IEEE Int. Conf. Acoustics Speech and Signal Processing（ICASSP2003），Hong Kong, China, 2003. (2):193 – 196
13. D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. Digital Signal Processing. 2000, (10):19-41
14. Zhenyu Xiong, Thomas Fang Zheng, Zhanjiang Song, and Wenhu Wu. Combining Selection Tree with Observation Reordering Pruning for Efficient Speaker Identification Using GMM-UBM. Proc. ICASSP. 2005, 625-628
15. http://www.nist.gov/speech/tests/spk/2002/resource/index.htm
16. Jean-FranÇois Bonastre, Sylvain Meignier, Teva Merlin. Speaker detection using multi-speaker audio files for both enrollment and test. In ICASSP 2003, Hong Kong, China

Design of Cubic Spline Wavelet for Open Set Speaker Classification in Marathi

Hemant A. Patil[1] and T.K. Basu[2]

[1] Department of Electronics and Instrumentation Engineering, Dr. B. C. Roy Engineering College, Durgapur, West Bengal, India
[2] Department of Electrical Engineering, Indian Institute of Technology,
IIT Kharagpur, West Bengal, India-721302
{hemant, tkb}@ee.iitkgp.ernet.in

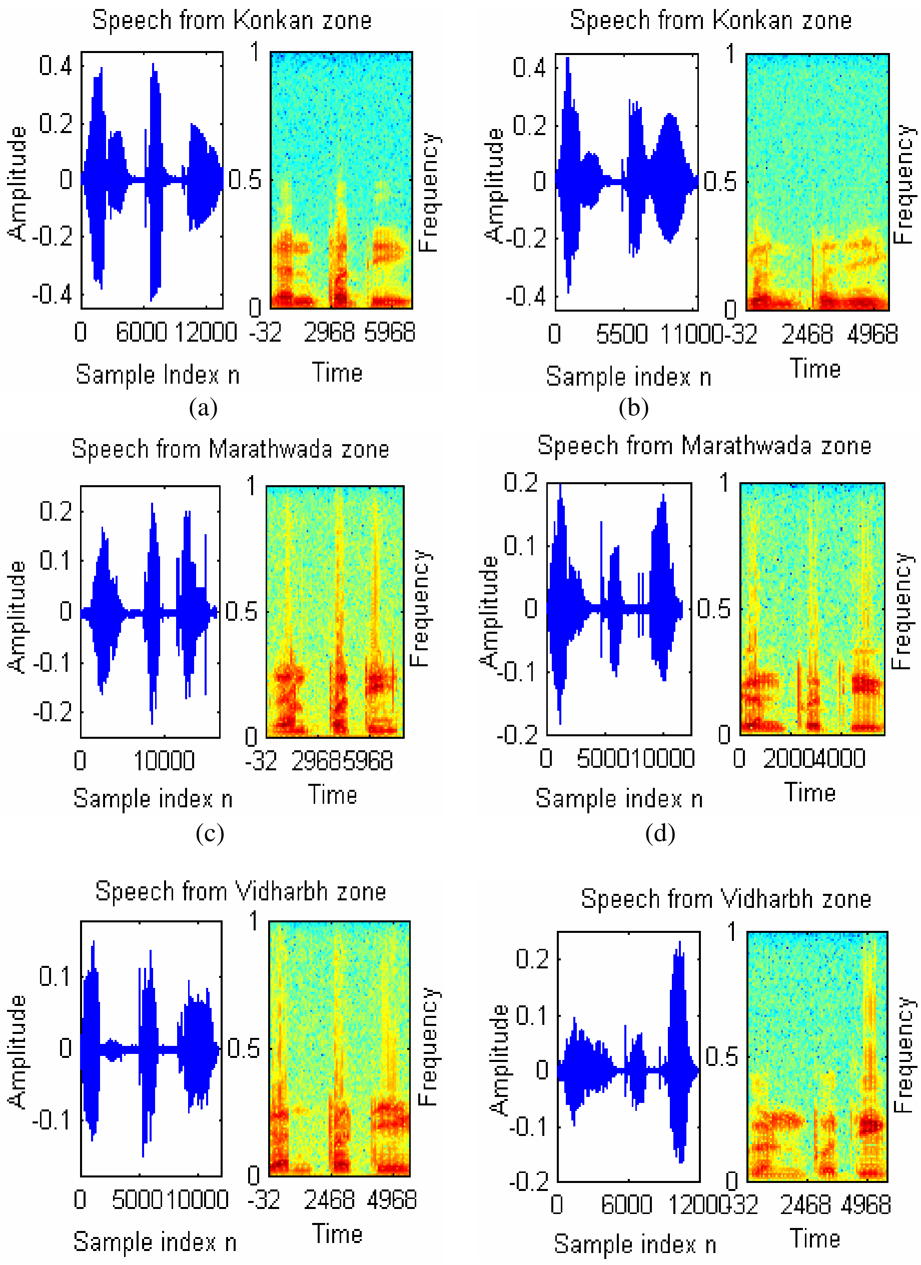
Abstract. In this paper, a new method of feature extraction based on design of cubic spline wavelet has been described. *Dialectal zone* based speaker classification in Marathi language has been attempted in the open set mode using polynomial classifier. The method consists of dividing the speech signal into nonuniform subbands in approximate Mel-scale using an admissible wavelet packet filterbank and modeling each dialectal zone with the 2nd and 3rd order polynomial expansions of feature vector. Confusion matrices are also shown for different dialectal zones.

1 Introduction

The problem of *speaker classification* can be defined in different ways [7]. In this paper, the problem of speaker classification is viewed from standpoint of automatic speaker recognition (ASR) [8], [13]. We define SC as grouping of the speakers residing in a particular dialectal zone based on their similar acoustical characteristics of speech. Such biometrics problem may be useful in *forensic science* applications such as in identifying a criminal's place of origin or in anthropological study of social ethnic group. Speakers residing in a particular dialectal zone will have *similar* dynamic use-patterns for their articulators which will be reflected in their spectrograms. Thus, if we bring an infant from zone Z_1 and bring him up in zone Z_2, then at an adult stage he will have articulators use pattern similar to that of zone Z_2 but not the zone Z_1. Fig. 1 shows speech corresponding to the word, *'Ganpati'*, (chosen because it has nasal-to-vowel coarticulation and hence it is highly speaker and possibly zone specific) spoken by two rural males from each of Konkan, Marathwada and Vidharbha zones. It is clear that the speech spectrograms of males belonging to same dialectal zones are similar if not identical whereas there are distinct dialectal differences in speech spectrograms of males from different zones.

ASR task can be performed in closed set or open set mode. In closed set, the unknown speaker to be identified/ classified is known to the machine whereas in open set scenario, the unknown speaker is not known to the machine which creates a

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 126 – 137, 2006.



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 1. Speech signal and its spectrogram corresponding to the Marathi word, “Ganpati”, spoken by rural males of (a) and (b) for Konkan, (c) and (d) for Marathwada , (e) and (f) for Vidharbha zones having age 36,51,35,35 27, and 34 respectively

challenge in assigning an unknown speaker's speech to its correct model. Hence, open set ASR is relatively much more difficult than that of closed set ASR. In this paper, the problem of open set speaker classification is addressed in *text-independent* mode by using a new feature set based on sub-band cepstrum on the database prepared in realistic noisy environments from three distinct dialectal zones of Maharashtra, *viz.*, Konkan, Vidharbha and Marathwada in an Indian language, *viz.*, Marathi.

2 Data Collection and Corpus Design

Database of 126 speakers is created from the three distinct dialectal zones of Maharashtra with the help of a voice activated tape recorder (*Sanyo* Model M-1110C & *Aiwa* JS299) with microphone input, a close talking microphone (*viz., Frontech* and *Intex*). Out-of these 126 speakers, 42 were from each of the dialectal zone. From 42 speakers in each zone, 21 speakers were used for machine learning and remaining 21 speakers as unknown to machine, i.e., success rate of the system is found with these speakers only. The recordings of the unknown and known speakers are done with different microphone which is more realistic condition in the *forensic* voice identification where the suspect's voice may be recorded with any other microphone. The data is recorded on the Sony high fidelity voice and music recording cassettes (C-90HFB). A list consisting of five questions, isolated words, digits, combination-lock phrases, read sentences and a contextual speech of considerable duration was prepared. The contextual speech consisted of description of nature or memorable events etc. of community or family life of the speaker. The data was recorded with 10 repetitions except for the contextual speech. During recording of the contextual speech, the interviewer asked some questions to speaker in order to motivate him to speak on his chosen topic. This also helps the speaker to overcome the initial nervousness and come to his natural mode so that the acoustic characteristics of his speech are tracked precisely. The speaker's voice and interviewer's voice were recorded on the same track. Once the magnetic tape was played into the computer, the speaker's voice was played again to check the wrong editing. The interviewer's voice was deleted from the speech file so that the models of the actual dialectal zone only can be made. The automatic silence detector was employed to remove the silence periods in the speech recordings to get only the models of a particular speaker and not the background noise and silence interval. Also, each speaker's voice is normalized by the peak value so that the speech amplitude level is constant for all the speakers in a zone. Finally, corpus is designed into training segments of 60s, 90s and 120s durations and testing segments of 1s, 3s, 5s, 7s, 10s, 12s and 15s durations in order to find the performance of the system for various training and testing durations [13].

3 Sub-band Based Cepstral Coefficients (SBCC)

Even though MFCC is extensively used for speaker recognition, it has got some drawbacks [4], [12], [13]:

- In MFCC, the filterbank is implemented with triangular filters whose frequency response is not smooth and hence may not be suitable for noisy speech data.

- The implementation of triangular filterbank requires critical band windowing (in frequency domain) or critical band filter banks (in time domain) which are computationally expensive as it does not involve any multirate processing.
- For computing the spectrum, DFT whose resolution is constant in time and frequency is used in MFCC. The local changes in time frequency plane will therefore not be highlighted very much in MFCC; this in turn will give less inter-zonal variability. Thus, speaker classification may not be satisfactory.

This motivated the authors, to investigate a new feature set which uses a theoretical framework similar to MFCC, i.e., extracting spectrum information in an approximate Mel scale and having the filters whose frequency response is smooth. In the next subsection, an elementary introduction to wavelet transform and wavelet packet transform followed by computational details of SBCC is presented.

3.1 Wavelet and Wavelet Packet Transform

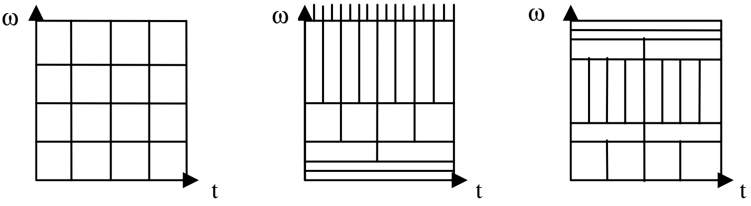
In continuous time, the wavelet transform is defined as the inner product of a signal $x(t)$ with a wavelet basis $\psi_{u,s}(t)$ in which the basis functions are scaled (by s) and translated (by u). The prototype wavelet is called as *mother wavelet* $\psi(t) \in L^2(\mathbb{R})$ [9], [12] (Lebesgue or Hilbert space of square integrable, i.e., finite energy functions) having zero average and unit norm so that

$$Wx(u,s) = \int_{-\infty}^{+\infty} x(t)\psi^*_{u,s}(t)dt \tag{1}$$

where $\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right)$

The discrete-time implementation of continuous-time wavelet transforms (CWT) given by (1), is achieved by using Mallat's algorithm. The continuous-time wavelet bases with the QMF (Quadrature Mirror Filter) banks used in discrete multirate processing decomposes the signal into different frequency bands essentially in the lower frequency side. But for speech processing applications, we need to decompose the higher frequency side also (in order to closely approximate the Mel scale) which motivates us to go for wavelet packets.

Wavelet packets were introduced by Coifmann, Meyer and Wickerhauser [2] by generalizing the link between multiresolution approximations and wavelet bases. A signal space V_j of a multiresolution approximation is decomposed in a lower resolution space V_{j+1} plus a detail space W_{j+1}.This is achieved by dividing the orthogonal basis $\left\{\phi_j(t-2^j n)\right\}_{n\in\mathbb{Z}}$ of V_j into two new orthogonal bases $\left\{\phi_{j+1}(t-2^{j+1}n)\right\}_{n\in\mathbb{Z}}$ of V_{j+1} and $\left\{\psi_{j+1}(t-2^{j+1}n)\right\}_{n\in\mathbb{Z}}$ of W_{j+1} where $\psi(t)$ and $\phi(t)$ are wavelet and scaling function respectively. Fig. 2 shows the time-frequency atoms of the basis functions for the Fourier, wavelet and wavelet packet transform. For MFCC, DFT is computed [3], whereas for SBCC, wavelet packet (WP) transform is computed. As DFT samples the Fourier transform at equally spaced points in the frequency domain and also the window length is constant throughout the analysis, the Heisenberg box for Fourier basis has constant time and frequency resolution, whereas WT and WPT, are computed via a time domain filtering with a down sampling by 2 (which approximates the scaling operation in continuous time and hence the variation in window length with frequency content of speech) which results in the variable time-frequency resolution in the time-frequency plane. If we consider a sub-band l, depending on the local changes in the DFT of the frame in the band, DFT coefficients will be different for different dialectal zones (or different speakers). But the subband signals are relatively robust to local changes within a band, as they reflect the overall spectral shape for a band. Hence, the WT and WPT, reflect the global changes in Fourier transform, while being relatively immune to local changes within a particular subband which may give less intra-zonal variability and hence improved performance for speaker classification. The decomposition for WP can be implemented by using a pair of QMF filter bank which divides the frequency band into equal halves.



(a) (b) (c)

Fig. 2. Time-frequency tilling (a) Fourier basis, (b) wavelet basis and (c) wavelet packet basis

Due to the decomposition of the approximation space (low frequency band) as well as the detail space (high frequency band), the frequency division of speech on both lower and higher side takes place. This recursive splitting of vector spaces is represented by an admissible WP binary tree. Let each subspace in the tree be represented by its depth *j* and number of subspaces *p* below it. The two wavelet packet orthogonal bases at a parent node (j, p) are defined by [9],

$$\psi_{j+1}^{2p}(t)=\sum_{n=-\infty}^{n=+\infty} h(n)\psi_j^p(t-2^j n) \quad \text{and} \quad \psi_{j+1}^{2p+1}(t)=\sum_{n=-\infty}^{n=+\infty} g(n)\psi_j^p(t-2^j n).$$

As $\left\{\psi_j^p(t-2^j n)\right\}_{n\in\mathbb{Z}}$ is orthonormal, $h(n)=\left\langle\psi_{j+1}^{2p}(v),\psi_j^p(v-2^j n)\right\rangle$ and $g(n)=\left\langle\psi_{j+1}^{2p+1}(v),\psi_j^p(v-2^j n)\right\rangle$.

As proposed by Farooq and Datta, the problem of selection of the best basis is solved by selecting a fixed set of basis which results into a fixed partitioning of the frequency axis such that it represents speech spectrum into the perceptually meaningful scale, i.e., the Mel scale [5] and [6]. The tree which has been selected in this paper is shown in Fig. 3. The implementation of SBCC is similar to that of MFCC, i.e., we pass the speech signal through the process of frame blocking, Hamming windowing, pre-emphasis and finally decomposing the speech into admissible wavelet packet structure, then finding the normalized filterbank energy (to have equal emphasis in each sub-band) and finally decorrelate the log-filterbank energy using DCT.

$$SBCC(k) = \sum_{l=1}^{L} \log\left(S(l)\right)\cos\left(\frac{k(l-0.5)}{L}\pi\right), k = 1, 2, \ldots\ldots, Nc.$$

L = Number of subbands in WP tree

N_c = Number of SBCC.

$SBCC(k)$ = k^th SBCC.

$S(l)$ = Normalized filter bank energy

i.e., $S(l) = \frac{\sum_{m\in l} \left[Wx(l,m)\right]^2}{N_l}$, N_l =Number wavelet coefficients in i^{th} subband.

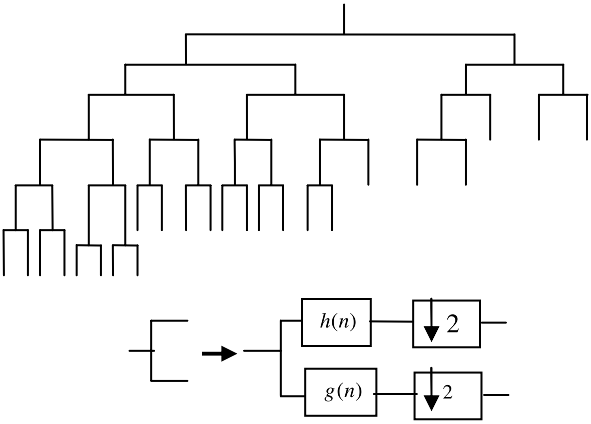


Fig. 3. 24 sub-band wavelet packet tree

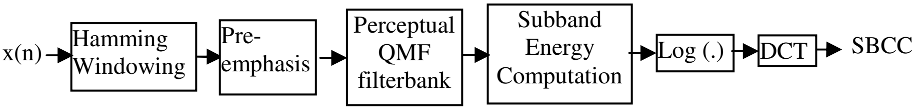


Fig. 4. Block diagram for SBCC and WPCC implementation

3.2 Spline Wavelet Design

In this paper, cubic spline wavelet has been designed for SBCC implementation. If $\{V_j\}_{j\in\mathbb{Z}}$ is a sequence of closed subspaces of $L^2(\mathbb{R})$ then multiresolution *causality* property imposes that $V_{j+1} \subset V_j$. In particular, $2^{-1/2}\phi(t/2) \in V_1 \in V_0$. Since $\{\phi(t-n)\}_{n\in\mathbb{Z}}$ is an orthonormal basis of V_0, we can decompose

$$\frac{1}{\sqrt{2}}\phi\left(\frac{t}{2}\right) = \sum_{n=-\infty}^{+\infty} h(n)\phi(t-n), \tag{2}$$

with $h(n) = \left\langle \frac{1}{\sqrt{2}}\phi\left(\frac{t}{2}\right), \phi(t-n) \right\rangle$.

(2) is called as scaling equation and it relates a dilation of ϕ by 2 to its integer translations. The sequence $h(n)$ will be interpreted as a discrete lowpass filter. In Fourier domain (2) can be expressed as

$$\Phi(2\omega) = \frac{1}{\sqrt{2}} H(\omega)\Phi(\omega) \Rightarrow H(\omega) = \frac{\sqrt{2}\Phi(2\omega)}{\Phi(\omega)} \tag{3}$$

The corresponding wavelet $\psi(t)$ has a Fourier transform defined by [9],

$$\Psi(\omega) = \frac{1}{\sqrt{2}} G\left(\frac{\omega}{2}\right)\Phi\left(\frac{\omega}{2}\right) \tag{4}$$

A box spline of degree m is a translation of $m+1$ convolutions of $\mathbf{1}_{[0,1]}$ with itself. It is centered at $t = 1/2$ if m is even and at $t = 0$ if m is odd. Its Fourier transform is given is,

$$\Phi(\omega) = \left[\frac{\sin(\omega/2)}{\omega/2}\right]^{m+1} \exp\left(\frac{-j\varepsilon\omega}{2}\right) \text{ with } \varepsilon = \begin{cases} 1 & m = even \\ 0 & m = odd \end{cases} \tag{5}$$

Using (5) in (3), we get

$$H(\omega) = \frac{\sqrt{2}\Phi(2\omega)}{\Phi(\omega)} = \sqrt{2}\left[\frac{\sin(\omega)}{\sin(\omega/2)}\right]^{m+1} \exp\left(\frac{-j\varepsilon\omega}{2}\right)$$

After trigonometry,

$$H(\omega) = \sqrt{2}\left[\cos(\omega/2)\right]^{m+1} \exp\left(\frac{-j\varepsilon\omega}{2}\right)$$

Now we construct a wavelet that has one vanishing moment by choosing $G(\omega) = O(\omega)$ (so that $G(\omega)$ has one zero at $\omega = 0$) in the neighborhood of $\omega = 0$. For example let us assume

$$G(\omega) = -j\sqrt{2}\sin(\omega/2)\exp(-j\varepsilon\omega/2)$$

Then the Fourier transform of resulting wavelet is given by (4),

$$\Psi(\omega) = \frac{1}{\sqrt{2}} G\left(\frac{\omega}{2}\right)\Phi\left(\frac{\omega}{2}\right) \Rightarrow \Psi(\omega) = \frac{-j\omega}{4}\left[\frac{\sin(\omega/4)}{\omega/4}\right]^{m+2} \exp\left(\frac{-j\omega(1+\varepsilon)}{4}\right)$$

It is the first derivative of a box spline wavelet from the box spline of degree $m+1$ centered at $t=(1+\varepsilon)/4$. For $m=3$, we get

$$\Phi(\omega)=\left[\frac{\sin(\omega/2)}{\omega/2}\right]^4 \text{ and } \Psi(\omega)=\frac{-j\omega}{4}\left[\frac{\sin(\omega/4)}{\omega/4}\right]^4 \exp\left(\frac{-j\omega}{2}\right)$$

$$H(\omega)=\left[\cos(\omega/2)\right]^4=\frac{H(z)}{\sqrt{2}}=\left(\frac{1+z^{-1}}{2}\right)^3 z^2$$

$$\Rightarrow \frac{h(n)}{\sqrt{2}}=\left\{0.0625, 0.25, \underset{\uparrow}{0.375}, 0.25, 0.0625\right\} \tag{6a}$$

Similarly,

$$G(\omega)=-j\sqrt{2}\sin(\omega/2)\exp(-j\varepsilon\omega/2)\Rightarrow\frac{G(z)}{\sqrt{2}}=-\left(\frac{1-z^{-1}}{2}\right)$$

$$\Rightarrow \frac{g(n)}{\sqrt{2}}=\left\{0.5, \underset{\uparrow}{-0.5}\right\} \tag{6b}$$

The above lowpass and highpass filter coefficients are used in SBCC computations and this completes the design of analysis filters for cubic spline wavelet. From (6) it is clear that $h(n)$ and $g(n)$ are symmetric odd length and anti-symmetric even length FIR filters having *linear phase*.

4 Experimental Results

In this paper, polynomial classifiers of 2nd and 3rd order approximations are used as the basis for all the experiments. Due to *Weierstrass-Stone approximation theorem,* polynomial classifiers are universal approximators to the optimal Bayes classifier [1]. The present work proposes a new feature set based on subband cepstrum by utilizing smooth cubic spline wavelet basis functions in wavelet packet transform for the open set speaker classification.

Feature analysis was performed using a 23.2 ms duration frame with an overlap of 50%. Hamming windows was applied to each frame and subsequently, each frame was pre-emphasized with the filter (1-0.97z-1). Pre-emphasis is smooth high pass filtering process applied to each speech frame which emphasizes high frequency components and de-emphasizes low frequency components, i.e., sharp/sudden changes in articulation are boosted up. This is also used to remove the effect of transfer function of glottis and thereby track changes solely related to vocal tract. Thus, pre-emphasis helps us to concentrate on *articulator dynamics* in speech frame and it is hence useful for tracking the *manner* in which the speaker pronounces a word. During training phase, 12 MFCC and 12 SBCC feature vectors were extracted per frame from the training speech as per the details discussed in the paper. SBCC were extracted with cubic spline wavelets discussed in section 3. These 12 dimensional feature vectors are fed to the classifier for model training. The classifier builds up model for each dialectal zone for different training durations such 60s, 90s, and 120s by averaging polynomial coefficients of the feature vectors of 21 speakers for each zone. During testing phase, 12 MFCC and 12 SBCC feature vectors were extracted per frame from the testing speech and score for each unknown speaker is computed against stored models of each dialectal zone. Finally, an unknown speaker is assigned to a zone whose score gives maximum value. Success rates for two dialectal zones, *viz.*, Konkan and Marathwada are shown in Tables 1-4 for different training (TR) and testing (TE) durations along with average success rates (over testing speech durations) in 8^{th} row for each table whereas in Tables 5-8 only average success rates over testing speech durations are shown for MFCC and SBCC and for 2 and 3 dialectal zones with 2^{nd} and 3^{rd} order polynomial approximation. Tables 9-16 show confusion matrix (diagonal elements show % correct classification in a dialectal zone and off-diagonal elements show the miss-classification) for Konkan (KN), Marathwada (MW) or Vidharbha (V). In tables 9-16, ACT represents actual dialectal zone of an unknown speaker and CLASS classified zone of an unknown speaker.

Table 1. Success Rates (%) for MFCC with 2^{nd} Order Approximation for 2 zones

TEST (SEC)	60s	90s	120s
1	50	69.04	78.57
3	50	69.04	73.81
5	50	66.66	73.81
7	50	64.28	76.19
10	50	71.42	78.57
12	50	71.42	76.19
15	50	71.42	78.57
Av.	**50.00**	**69.04**	**76.53**

Table 2. Success Rates (%) for MFCC with 3^{rd} Order Approximation for 2 zones

TEST (SEC)	60s	90s	120s
1	100	100	100
3	100	100	100
5	100	100	100
7	100	100	100
10	100	100	100
12	100	100	100
15	100	100	100
Av.	**100**	**100**	**100**

Table 3. Success Rates (%) for SBCC with 2^{nd} Order Approximation for 2 zones

TEST (SEC)	60s	90s	120s
1	95.23	95.23	95.23
3	100	100	100
5	100	100	97.61
7	100	100	100
10	97.61	100	97.61
12	97.61	97.61	97.61
15	100	100	100
Av.	**98.63**	**98.97**	**98.29**

Table 4. Success Rates (%) for SBCC with 3^{rd} Order Approximation for 2 zones

TEST (SEC)	60s	90s	120s
1	95.23	97.61	95.23
3	100	100	100
5	100	100	100
7	100	100	100
10	100	100	100
12	100	100	100
15	100	100	100
Av.	**99.31**	**99.65**	**99.31**

Table 5. Success Rates with 2nd Order Approximation for 2 zones

TR FS	60s	90s	120s
MFCC	**50.00**	**69.04**	**76.53**
SBCC	**98.63**	**98.97**	**98.29**

Table 6. Success Rates with 3rd Order Approximation for 2 zones

TR FS	60s	90s	120s
MFCC	**100**	**100**	**100**
SBCC	**99.31**	**99.65**	**99.31**

Table 7. Success Rates with 2nd Order Approximation for 3 zones

TR FS	60s	90s	120s
MFCC	**39.67**	**40.58**	**47.38**
SBCC	**69.15**	**71.87**	**71.19**

Table 8. Success Rates with 3rd Order Approximation for 3 zones

TR FS	60s	90s	120s
MFCC	**79.36**	**81.40**	**82.99**
SBCC	**71.65**	**71.65**	**73.23**

Table 9. Confusion Matrix for MFCC with 2nd Order Approximation for 2 zones

ACT. CLASS.	KN	MW
KN	**57.14**	42.85
KN	0	**100**

Table 10. Confusion Matrix for MFCC with 3rd Order Approximation for 2 zones

ACT. CLASS.	KN	MW
KN	**100**	0
KN	0	**100**

Table 11. Confusion Matrix for SBCC with 2nd Order Approximation for 2 zones

ACT. CLASS.	KN	MW
KN	**100**	0
KN	0	**100**

Table 12. Confusion Matrix for SBCC with 3rdOrder Approximation for 2 zones

ACT. CLASS.	KN	MW
KN	**100**	0
KN	0	**100**

Table 13. Confusion Matrix for MFCC with 2nd Order Approximation for 3 zones

ACT. CLASS.	KN	MW	V
KN	**42.85**	0	57.14
MW	0	**0**	100
V	0	0	**100**

Table 14. Confusion Matrix for MFCC with 3rd Order Approximation for 3 zones

ACT. CLASS.	KN	MW	V
KN	**100**	0	0
MW	0	**100**	0
V	0	42.85	**57.14**

Table 15. Confusion Matrix for SBCC with 2nd Order Approximation for 3 zones

ACT. CLASS.	KN	MW	V
KN	**100**	0	0
MW	19.04	**80.95**	0
V	19.04	57.14	**23.81**

Table 16. Confusion Matrix for SBCC with 3rd Order Approximation for 3 zones

ACT. CLASS.	KN	MW	V
KN	**100**	0	0
MW	9.523	**90.47**	0
V	4.761	80.95	**14.28**

Some of the observations from the results are as follows:

- For 2nd order approximation, proposed feature set SBCC (extracted with cubic spline wavelet) outperforms MFCC in almost all the cases of training and testing speech durations whereas for 3rd order approximation MFCC performs slightly better than SBCC.
- For 2nd order approximation and SC task for 3 zones, confusion matrix for SBCC performed relatively better than MFCC whereas for 3rd order approximation, MFCC performed well than MFCC.
- Training and testing speech durations influence the success rates in majority of the cases.
- Average success rates increase with the increase in training speech durations and hence for good performance, one should train the system with speech of more than 1 min. durations.
- On the whole, proposed feature set performs better than MFCC in majority of the cases. This may be due to the fact that the spline wavelets are linear-phase and smooth wavelets and hence may be able to represent zone specific features more efficiently.

5 Conclusion

In this paper, a new feature set based on subband cepstrum is proposed for the problem of speaker classification. The spline wavelet is designed and employed in SBCC computation. The performance of newly proposed feature set was compared with MFCC and found to be effective in majority of the cases. The investigations carried out in this paper can be of significant importance in *forensic acoustics*.

References

1. Campbell, W. M., Assaleh, K. T., Broun, C. C.: Speaker recognition with polynomial classifiers. IEEE Trans. on Speech and Audio Processing. 10 (2002) 205-212
2. Coifman, R. R., Meyer, Y., Wickerhauser, M. V.: Wavelet analysis and signal processing. Wavelets and Applications. Boston, Jones and Bartlett. B. Ruskai et al. editor, (1992)153-178
3. Davis, S. B., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust., Speech and Signal Processing. 28 (1980) 357-366
4. Erzin, E., Cetin, A. E., Yardimci, Y.: Subband analysis for robust speech recognition in the presence of car noise. Proc. of Int. Conf. on Acoust., Speech and Signal Processing. 1(1995)417-420
5. Farooq, O., Datta, S.: Mel filter-like admissible wavelet packet structure for speech recognition. IEEE Signal Processing Letters. 8(2001)196-198
6. Farooq, O.: Wavelet: a new tool for speech recognition. Proc. of the Int. Workshop on Technology Development in Indian Languages. 2003
7. Jin, H., Kubala, F., Schwartz, R.: Automatic Speaker Clustering. Proceedings of the Speech Recognition Workshop. (1997)108-111

8. Kersta, L.G.: Voiceprint Identification. Nature 196(1962)1253-1257
9. Mallat, S.: A Wavelet Tour of Signal Processing. 2nd Edition, Academic Press, 1999
10. Mitra, Snehesh, Patil, Hemant A., Basu, T. K.: Polynomial classifier techniques for speaker recognition in Indian languages. Proc. National System Conference. IIT Kharagpur, India, (2003)304-308
11. Sarikaya, R., Pellon, B. L., Hansen, J. H. L.: Wavelet packet transforms features with application to speaker identification. IEEE Nordic Signal Processing Symposium. (1998)81-84
12. Sarikaya R., Hansen, J. H. L.: High resolution speech feature parameterization for monophone based stressed speech recognition. IEEE Signal Processing Letters. (2000)182-185
13. Patil, H. A.: Speaker Recognition in Indian languages: A feature based approach. Ph.D. thesis, department of electrical engineering, IIT Kharagpur, India, July 2005.

Rhythmic Organization of Mandarin Utterances — A Two-Stage Process

Min Chu[1] and Yunjia Wang[2]

[1] Microsoft Research Asia, Beijing
[2] Department of Chinese Language and Literature, Peking University, Beijing
minchu@microsoft.com, wangyunjia@pku.edu.cn

Abstract. This paper investigates the rhythmic organization of Mandarin utterances through both corpus analyses and experimental studies. We propose to add a new prosodic unit, the principle prosodic unit (PPU), into the prosodic hierarchy of Mandarin utterances. The key characteristic of PPU is that inner-unit words normally have to be spoken closely, while inter-unit grouping is rather flexible. Because of this, we further suggest that the rhythmic organization of Mandarin utterances is a two-stage process. In the first stage, syllables are grouped into prosodic words, and then to PPUs. The forming of PPUs is restricted by the local syntactic constraint and the length constraint. In the second stage, though the rhythmic constraint still has influences, the grouping of PPUs into phrases is rather flexible. Normally, multiple equally good solutions exist for a sentence in this stage.

Keywords: prosodic hierarchy, prosodic phrasing, principle prosodic unit, syntactic constraint, length constraint.

1 Introduction

Breaking utterances into prosodic units is an important part of speech production. Therefore, for a text-to-speech (TTS) system, properly prosodic phrasing is crucial for achieving high naturalness. Various stochastic models have been used in predicting prosodic constituents from text [1][2][3][4][5]. Two types of features, the syntax related features (such as the syntactic trees or part-of-speech of succeeding words *etc*) and length related features (include sentence length, the distance from the beginning and the end of the sentence, the distance from previous breaks *etc*), are normally used in such predictions. Since there is no commonly accepted specification on how to label prosodic phrases, more phrase boundaries are labeled in some training corpus than in the others. Therefore, the prediction accuracy of different models trained with different corpus cannot be compared directly. However, when putting various results together, we still see two common problems. On the one hand, it seems to have a ceiling for the prediction accuracy. The accuracy of single level break prediction is between 85%-95%, and the accuracy of multi-level prediction is between 80%-85% [1].

[1] The accuracy of the prediction single break boundary various greatly in different works. One important reason is that the definition of boundary changes. In some data, only a few breaks are labeled and most of them go with punctuations. Then the accuracy is higher. In other works, minor breaks are considered. And the accuracy is lower.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 138 – 148, 2006.

On the other hand, although some phrase boundaries generated by the prediction model do not aligned with the labeled boundaries in the testing samples, they are judged as acceptable breaks by the human listeners. Though many works are being carried on to increase the accuracy by incorporating new features or better prediction models, we believe the space for such improvements is limited because natural variations in prosodic phrasing has not been considered.

In this paper, the variations in prosodic phrasing within Mandarin utterances are investigated in a parallel speech corpus where three repetitions are recorded for 1000 sentences. Both variable and invariable prosodic constituents are observed. Two low level prosodic units, the prosodic word and the principle prosodic unit, are found to have rather stable segmentations in the three repetitions. While the high level units, the prosodic phrase and the intonational phrase, are found to have large variations. These observations lead us to propose a two-stage decision process in the rhythmic organization of Mandarin utterances. In the first stage, syllables are first grouped into the prosodic words and then into the principle prosodic units under both syntactic constraints and length constraints restrictedly. In the second stage, the prosodic principle units are grouped into phrases rather freely. Although the length constraint still reacts on the forming of prosodic phrases, we proof with a perceptual experiment that the syntactic structure of the sentence does not play an important role in this stage.

2 Variable and Invariable Prosodic Constituents in Mandarin

There are many studies on specifying prosodic hierarchies. Syllable is often viewed as the lowest unit of prosodic constituent structure. According to Selkirk [6][7], the suprasyllabic hierarchy for English includes at least four categories: the foot, the prosodic word, the phonological phrase and the intonational phrase. The foot is usually smaller in size than the word and has been used in representing the distinction between stressed and unstressed syllables. The prosodic word is a roughly word-sized unit and it is required particularly when the words defined in syntactic terms fail to correspond exactly to the “words” playing a role in prosody. The intonational phrase is the highest prosodic unit within an utterance and tends to correspond to a simple sentence without extrapositions or interruptions. The phonological phrase is a constituent falling between the intonational phrase and the prosodic word. In some works, the intonational phrase is also referred as the major phrase and the phonological phrase is named as the prosodic phrase, the intermediate phrase or the minor phrase [4, 8].

In the prosody studies of Mandarin [5][9][10], a three-tier hierarchy, which includes the prosodic word, the prosodic phrase and the intonational phrase, has been widely adopted. Here, the definitions of prosodic and intonational phrases are rather similar to those used in English. Yet, the term prosodic word is often used interchangeable with the term prosodic foot. Both are used to describe the predominance of the disyllabic pattern of bottom rhythmic unit in Mandarin. According to [11], disyllabic prosodic feet (or prosodic words) are widely used in Mandarin speech, trisyllabic ones are acceptable, but, the using of monosyllabic or quadrisyllabic feet are constrained to limited situations.

2.1 The Parallel Speech Corpus

A Mandarin speech database that contains three repetitions of 1,000 sentences, denoted as HF1, HF2 and ZT, respectively, is used in this study. HF1 and HF2 were recorded by the same voice talent, separated by a time span of 6 months. ZT was recorded by another voice talent. The speakers were instructed to read these sentences with unmarked style. Therefore, we assume that no intentional change in the linguistic and affective expression among the three repetitions. Based on this assumption, differences in prosodic phrasing among HF1, HF2 and ZT are viewed as natural variations in unmarked reading speech.

Four levels of break indices (B1-B4) were labeled manually by listening to the speech, among which B2, B3 and B4 correspond to weak, moderate and strong break, respectively [12]. Although the break indices were annotated perceptually, they have a rough correspondence to the prosodic hierarchy of Mandarin. B2 corresponds to a prosodic phrase (or a minor phrase) boundary. Both B3 and B4 mark intonation phrase (or major phrase) boundaries, but B4 is followed by a longer pause. B1 is the prosodic word boundary. Since the prosodic word is a bottom rhythmic unit and does not have noticeable marks in speech, its boundary is annotated with rules shown in Table 1.

Table 1. Rules for annotating prosodic word boundaries

Rule 1	When a disyllabic or trisyllabic word doesn't have a clitic and a monosyllabic word attached, it forms a prosodic word by itself; otherwise, it forms a prosodic word together with the clitic or the proceeding or succeeding monosyllabic word.
Rule 2	A monosyllabic word is grouped into a prosodic word together with the word either before or after it unless the monosyllabic word is significantly lengthened or separated from both the proceeding and succeeding words.
Rule 3	All words with more than three syllables should be decomposed into a series of disyllabic or trisyllabic prosodic words. The clitic or mono-syllabic word before or after it should be merged into the first or last prosodic word.

The corpus was annotated by three well-trained annotators. For prosodic word boundaries, the ratio of all three annotators agreeing with each other is 96.6% and the ratio of at least two agreeing is 99.9%. For the 4 level break indices, the ratio of all agreeing is 82.9% and the ratio of at least two agreeing is 99.1%. The final break indices are generated by voting. For a few cases that all three annotators disagree with each other, the middle level indices were chosen. Table 2 gives some examples of the break indices.

The acoustic analysis of the three level boundaries [13] shows that the both prosodic phrase and the intonational phrase are signaled by final lengthening and pause. Pitch resets are observed across intonational phrase boundaries, yet, not across prosodic phrase boundaries.

Table 2. Examples of the annotated break indices

1. 四家 B2 商业 B1 银行 B3 确定 B2 下半年 B2 贷款 B1 投向 B4。
2. 他B2扛起B1背包B3，深入B2基层B1消防队B4，每年B2下基层B3都在B2二百天B1以上B4。
3. 从B2经济B2和B1环保的B2角度B1看B4，天然气B2是B1最好的B1燃料B4。
4. 大事B2不糊涂B3，小事B2不介意B4。

2.2 Variations in Prosodic Constituents

The prosodic phrasing realized in an utterance is governed by phonological rules and paralinguistic factors such as speaking rate, speaking style, special intention or speaker's personal habit. It will be very difficult to analyze if all factors are activated together. Therefore, in this paper, we focus on the rhythmic organization in speech recorded with unmarked speaking style. The first question we want to answer is, given the same content, the same speaking style and the same speaking rate, is the prosodic phrasing the same?

First, we compared the observed prosodic structure of the same sentence recorded in HF1 and HF2. We found that about 14% syllables were grouped into different prosodic units. This shows that the difference in prosodic organization between the two repetitions of the same speaker is rather large. Then, we looked into differences in the three prosodic constituents.

Prosodic Word. We found that only 2.7% syllables were organized into different prosodic words, i.e. the organization of prosodic words is rather stable. The few differences are mainly related to the grouping of monosyllabic syntactic words. For example, the phrase "从经济和环保的角度看(viewing from the angle of economy and environment protection)", was read into four prosodic words as "从经济 | 和环保的 | 角度 | 看" in HF1. Yet, it was read into 6 prosodic words in HF2 as "从| 经济 | 和 | 环保的 | 角度 | 看" by lengthening the monosyllabic words "从 (from) " and "和 (and) ". The meanings in both readings are the same.

Prosodic Phrase. After excluding the unmatched prosodic words in HF1 and HF2, we compared again the organization of prosodic phrases. 12.8% of prosodic words were found been grouped into different prosodic phrases, i.e. more variations were observed in the organization of this unit. For example, the sentence "大事不糊涂小事不介意 (be clearly minded when dealing with important issues but not nitpicking on trivial matter) " was grouped into four prosodic words as "大事|不糊涂|小事|不介意" in both HF1 and HF2. The four prosodic words were grouped into two prosodic phrases in HF1 as "大事不糊涂 || 小事不介意", and each of them was treated as one prosodic phrase in HF2 as "大事 || 不糊涂 || 小事 || 不介意". Again, the meaning has not been changed.

Intonational Phrase. In order to compare the organization of intonation phrase, all unmatched prosodic phrases were excluded. In the remaining corpus, 22.5% prosodic phrases were found to be organized into different intonation phrases, i.e. larger variations were observed. For example, the phrase "每年下基层都在二百天以上 (Stay with the grass roots for more than 200 days per year)" were grouped into three prosodic phrases as “每年下基层 ‖ 都在‖二百天以上” in both HF1 and HF2. They were organized into one intonational phrase in HF1 and two intonational phrases in HF2 as “每年下基层 ⦀ 都在二百天以上”.

Then, recordings in HF1 and HF2 were compared with ZT, respectively. The differences in prosodic words are 2.4% and 0.9% respectively. The differences in prosodic phrases are 12.5% and 11.5% and the differences in intonational phrases are 17.5% and 33.5%. It is clear that the inter-speaker differences are not larger than the inner-speaker differences. Therefore, we conclude that the differences in the rhythmic organization among the three repetitions are not caused by personal habit but by the freedom in speech organization. The variability is larger in high level units than in low level ones.

At last, we looked into the matched and unmatched phrases in the three repetitions and found that some words that have certain types of syntactic relationship (such as the modifiers and the heads in some of the base adjunct-head phrases) are stably grouped into the same phrase. Yet, words with other relationships can be grouped into either the same phrase or different phrases. For example, in the sample sentence 1 in Table 2, “商业” and “银行”, “贷款” and “投向” are in the same phrase in all the three repetitions, but “四家” and “商业银行” are within the same prosodic phrase in HF1 and ZT, but are two prosodic phrases in HF2.

These observations imply the existence of another stable prosodic unit. We name it the *principle prosodic unit* (PPU). The key feature of the PPU is that, any break within it will make the speech sound unnatural or influent, while whether a PPU is grouped with its neighboring one(s) into a prosodic phrase is rather flexible. We believe that adding such a prosodic unit is helpful to explain the variations in prosodic phrasing.

2.3 Characteristics of the Principle Prosodic Unit

In order to describe the principle prosodic unit more specifically, we need a labeled corpus for analysis. Since the PPU is a unit that has no noticeable cues in its phonetic implementation, annotation of PPU directly in a text or speech corpus is not easy. Fortunately, we have the parallel corpus. We assume that prosodic word boundaries that haven’t been phonetically implemented as a phrase boundary in any of the three repetitions are no-break positions. Other word boundaries are positions that are allowed to have breaks. Then, any chunk that contains only no-break word boundaries forms a PPU. Table 3 lists examples of the PPU labels.

Table 3. Examples of the principle prosodic unit

(1) 四家 ‖ 商业银行 ‖ 确定‖下半年 ‖ 贷款投向。
(2) 他 ‖ 扛起背包 ‖，深入 ‖ 基层消防队 ‖，每年 ‖ 下基层 ‖ 都在‖ 二百天以上。
(3) 从 ‖ 经济 ‖ 和 ‖ 环保的 ‖ 角度看 ‖，天然气 ‖ 是 ‖ 最好的燃料。
(4) 大事 ‖ 不糊涂 ‖，小事 ‖ 不介意。

In the annotated corpus, we found that about 56% prosodic words are PPUs themselves and about 40% PPUs contains two prosodic words. Only 4% PPUs contain more than two prosodic words. Therefore, we believe that the most important operation in the rhythmic organization of Mandarin utterances is to decide whether two neighbored prosodic words have to be spoken without any pause. Furthermore, we found that the forming of PPUs was constrained mainly by two factors. One is the local syntactic relationship between two succeeding prosodic words. Only words that have the syntactic relation with high adhesive strength should be forced into the same PPU. The other is the length constraint, i.e. normally a PPU contains no more than two prosodic words.

By adding the PPU into the prosody hierarchy of Mandarin utterance, we have four prosodic units altogether. The organization of the two low-level units (the prosodic word and the principle prosodic unit) is rather stable in speech and is not affected much by who speaks, when and how it is spoken. Yet, the organization of the two high-level units (the prosodic phrase and the intonational phrase) is rather flexible, being governed by different paralinguistic or nonlinguistic factors.

3 The Two-Stage Process in Mandarin Rhythmic Organization

Based on above analysis, we would like to propose that the rhythmic organization of Mandarin utterances contains two stages.

In the first stage, syllables are first grouped into prosodic words and then PPUs under the local syntactic constraint and the length constraint. The process of building up PPUs in a sentence is illustrated in Fig. 1. First, a sentence is segmented into a sequence of prosodic words $w_1w_2w_3w_4.....w_N$ [2]. Any two succeeding words w_i and w_{i+1} have either a local syntactic relationship (such as modifier and noun head, modifier and verb head, *etc*) or have no syntactic relationship. Their syntactic relationship is demoted as $r_{i,i+1}$. For Mandarin, about 30 types of local syntactic relationships are found, each attached with an adhesive strength [15]. The adhesive strength shows how likely the two prosodic words should be in the same PPU. If the adhesive strength between two words is smaller than a threshold R, the two words will not be in the same PPU. For a word w_i in the sentence, if $r_{i,i+1}> r_{i-1,i}$ and $r_{i,i+1}>$ R, w_i will form a PPU with w_{i+1}; if $r_{i-1,i} > r_{i,i+1}$ and $r_{i,i+1} >$ R, it will form a PPU with w_{i-1}. Once two words w_i and w_{i+1} form a PPU, the PPU reaches it up-bound in length, therefore, their adhesive strength with neighbored words w_{i-1} and w_{i+2} decreases so that w_{i-1} and w_{i+2} normally will not be in the same PPU as w_i and w_{i+1}. PPUs don't have directly prosodic marks in speech. They only lay out candidates for prosodic phrasing.

In the second stage, PPUs are grouped into prosodic phrases and intonational phrases which are signaled with perceptible boundary signals, such as pauses, pitch contours or final lengthening at the phrase boundaries. We believe that the grouping is PPUs is rather flexible and even think that such a grouping is performed without considering the sentence syntax structure although many other factors may affect the final realization. Among other factors, the length-balance constraint is an important one, i.e. prosodic phrases in an utterance tend to be similar in lengths. To verify these ideas, we performed a perceptual experiment.

[2] There already have many studies on prosodic word segmentation [14]. In this study, we skip it and assume that we already have prosodic word sequence.

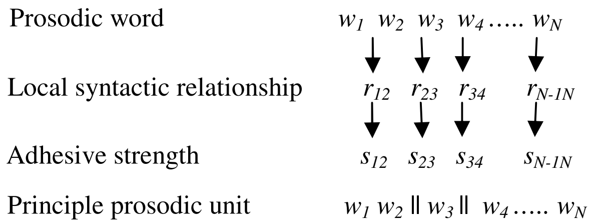

Prosodic word
w_1 w_2 w_3 w_4..... w_N
Local syntactic relationship
r_{12} r_{23} r_{34} r_{N-1N}
Adhesive strength
s_{12} s_{23} s_{34} s_{N-1N}
Principle prosodic unit
w_1 w_2 || w_3 || w_4..... w_N

Fig. 1. The formation of the principle prosodic unit

4 Perceptual Experiment

Two hypotheses are tested in the experiment: the first is that many ways to break an utterance are acceptable and breaks can be allocated to principle prosodic unit boundaries without consider the syntactic structure; the second is that the break allocation among PPU boundaries is constrained by the length-balance effect to some extent.

4.1 Design and Procedure of the Experiment

Twenty sentences with 13~19 syllables were selected from the parallel speech corpus. These sentences contain annotations of both the PPU and three real solutions of prosodic phrasing. Five different hypothetical phrasing solutions are generated automatically for each sentence. They all comply with the two hypothesis and are referred as positive samples P1~P5. Besides, two negative samples (N1 and N2) are generated for each sentence by putting a phrase boundary within a PPU and one golden solution (G1) is prepared by selecting one from the three real solutions. Examples for the eight phrasing solutions are given in Table 4. The 20 sentences were synthesized with Mulan TTS system [16] by using the eight different phrasing solutions. The differences among the synthetic utterances are assumed mainly caused by the differences in prosodic phrasing. The golden solutions (G1) draw the up-bound of the naturalness and negative solutions (N1 & N2) plot the bottom-bound. The aim of this experiment is to see if positive solutions (P1~P5) distributed in between.

There are at least two ways to compare the naturalness among the eight samples of each sentence. One is to ask subjects to directly score each utterance. The other is to rank them by one-to-one comparison. Since the main difference among the eight samples is the break positions, obtaining MOS scores precise enough to distinguish their naturalness is not easy. Therefore, we chose the second method, i.e. we compared the eight utterances to one another and asked the subjects to choose which utterance sounds better. 28 pairs of utterances were generated for each sentence and altogether 560 pairs were obtained for the 20 sentences.

20 university students participated in the experiment, each worked with a scoring tool in a standard PC. They listened to the utterance pairs through headphones and chose either "A sounds better" or "B sounds better".

Table 4. Rules for generating eight phrasing solutions and examples

Rules to generate the phrasing solution	Examples
P1: No break in the whole sentence	这个惊人的数字凝聚着一种民族精神。
P2: Only one break at the PPU boundary closest to the middle point of the sentence on the left	这个惊人的数字 | 凝聚着一种民族精神。
P3: Only one break at the PPU boundary closest to the middle point of the sentence on the right	这个惊人的数字凝聚着 | 一种民族精神。
P4: Two breaks at PPU boundaries closest to the 1/3 and 2/3 points of a sentence on the left[3]	这个 | 惊人的数字凝聚着 | 一种民族精神。
P5: Two breaks at PPU boundaries closest to the 1/3 and 2/3 points of a sentence on the right	这个惊人的数字 | 凝聚着一种 | 民族精神。
N1: One inner-PPU break at the left part of the sentence[4]	这个惊人的 | 数字凝聚着一种民族精神。
N2: One inner-PPU break at the right part of the sentence	这个惊人的数字凝聚着一种民族 | 精神。
G1: One from the parallel corpus	这个 | 惊人的数字 | 凝聚着一种 | 民族精神。

4.2 Results and Analysis

Measurement. If an utterance was judged as the better one in one comparison by one subject, it received one point. Otherwise, it got no points. The ratio between the points an utterance obtained and the number of times it was compared is defined as the *preference rate* (PR) of the utterance.

Preference rate of golden, positive and negative solutions. After the adjustment in utterances and subjects, PRs for golden, positive and negative phrasing solutions are calculated. As shown in Fig. 2, both golden and positive utterances sound significantly better than the negative utterances. The naturalness of positive utterances is in the same range of the golden ones. This result supports our first hypothesis. Breaks can only be allocated to PPU boundaries. It is not necessary to consider any syntactic information during the allocation. Many ways to break an utterance are equally good.

Length constraint in prosodic phrasing. Although most positive samples and golden samples sound similar in terms of their naturalness, some are worse than others. Nine samples with PRs lower than 0.3 are found. One of them is in the no-break category (P1), four in the single-break category (P2&P3) and four in the natural-break category (G1). None is in the two-break category (P4&P5). It seems that single-break and natural-break categories have more problems than others. Therefore, detail analyses are performed in the two categories.

[3] In P4 and P5, if no boundary on the left or right is found, the one on the right or left can be used. We have made sure that P4 and P5 differ at least in one break position.

[4] In N1 and N2, breaks are still put at prosodic word boundaries, but they are within a prosodic chunk.

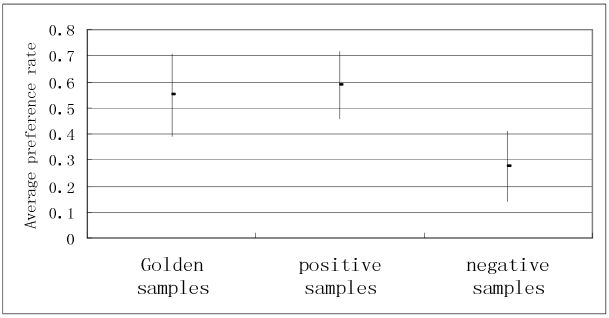


Fig. 2. PRs in golden, positive and negative utterances

In the single-break type, we found that a possible reason for the low PR is the large difference between the lengths of the two prosodic phrases. Therefore, a *length balance ratio* (LBR) is defined as the ratio between the length difference of the two prosodic phrases in an utterance and the average length of the two phrases. The relationship between LBR and PR of utterances is shown in Fig. 3. The solid line is the trend line. The correlation coefficient between the two parameters is -0.44. That is to say the PR of an utterance is negatively correlated to its LBR.

Among the four worst natural-break samples, three have breaks at all PPU boundaries. This implies that too many breaks may hurt the naturalness of synthesized speech although human speaker can do well with the same breaks. Therefore, the golden samples are further decomposed in accordance with the number of breaks. The utterance that has two breaks is merged into the two-break category. Eleven utterances have three breaks and six have four. They formed the two new categories. Fig. 4 shows the statistics of PRs in groups of utterances with 0-4 breaks. We find that utterances with two breaks have the highest PR and those with four breaks have the lowest PR. However, in an ANOVA analysis, these differences are not statistically significant. The possible reason is because we have too few samples in the three-break and four-break categories.

Both Fig. 3 and 4 support our second hypothesis, i.e. the break allocation among PPU boundaries are constrained by the length of prosodic phrases. Prosody phrases in an utterance tend to be similar in length.

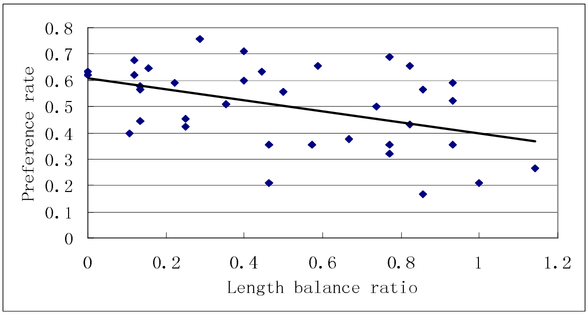


Fig. 3. LBR and PR of utterances

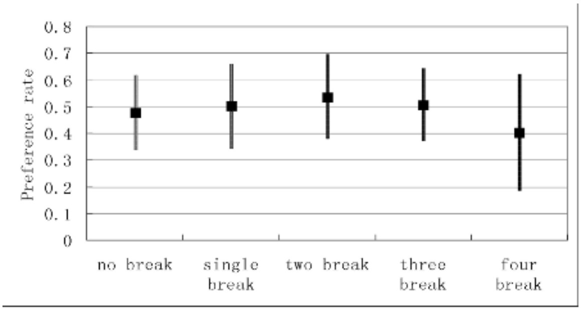


Fig. 4. PRs in groups of utterances with various numbers of breaks

5 Conclusion and Discussion

This paper investigates the variability in prosodic phrasing through both corpus analysis and a perceptual experiment. The results support our suggestion on the existence of a stable prosodic constituent, the principle prosodic unit, in Mandarin. On the one hand, breaks appear within a PPU will significantly hurt the naturalness of synthesized speech; on the other hand, many ways to group PPUs into prosodic phrases are equally good. Therefore, we propose that the rhythmic organization of Mandarin utterances is a two-stage process. In the first stage, syllables are organized into PPUs. In the second stage, PPUs are grouped into prosodic phrases. Syntactic constraints mainly affect the forming of PPUs but not that of the upper-level prosodic phrases. Length constraints play an important role in both stages.

Based on above conclusions, we can see that predicting no-break locations in a sentence is much more important than predicting the break locations in the prosodic phrasing in Mandarin. Therefore, the research focus on prosodic phrasing should adjust from predicting the best phrasing solution to predicting the no-breaking locations (or the PPU boundaries). Once having the PPU boundaries, many flexible ways can be used to group PPUs into phrases. For example, we can generate more phrases (or short phrases) in slow speech and fewer phrases (or longer phrases) in fast speech.

References

1. Wang, M.Q., Hirschberg, J.: Predicting Intonational Phrasing from Text, Proc. of Association for Computational Linguistics 29th annual meeting (1991) 285-292
2. Hirschberg, J., Prieto, P.: Training Intonatinal Phrasing Rules Automatically for English and Spanish Text-to-Speech, Speech Communication, Vol. 18 (1996) 281-290
3. Lee, S., Oh, Y. H.: Tree-Based Modeling of Prosodic Phrasing and Segmental Duration for Korean TTS Systems, Speech Communication, Vol. 28 (1999) 283-300
4. Ostendorf, M., Veilleux, N.: A Hierarchical Stochastic Model for Automatic Prediction of Prosodic Boundary Location, Computational Linguistics, Vol.20, No.1 (1994) 27-54

5. Chu, M., Qian, Y: Locating Boundaries for Prosodic Constituent in Unrestricted Mandarin Texts, Journal of Computational Linguistics and Chinese Language Processing, Vol.6. No.1. (2001) 61-82
6. Selkirk, E. O.: The Role of Prosodic Categories in English Word Stress, Linguistic Inquiry 11 (1980) 563-605
7. Selkirk, E. O.: On the Nature of Phonological Representation, in Amderspm. J., Laver, J. and Meyers, T. Eds., The Cognitive Representation of Speech, Amsterda: North Holland (1981)
8. Ladd, D. R.: Intonational Phonology, Cambridge University Press, Cambridge (1996)
9. Shen, M. X., Xu, B.: A CART Based Hierarchical Stochastic Model for Prosodic Phrasing in Chinese, Proc. of ISCSLP (2000)
10. Li, J.F., Hu, G.P., Fan, M., Dai. L.R.: Apply Length Distribution Model to Intonational Phrase Prediction, proc. of International Symposium on Chinese Spoken Language Processing (2004) 213-216.
11. Wang, H.J.: The Conjunction between Prosodic Boundaries in Mandarin and Rhythm Mode, Syntax and Pragmatics, Collection of Linguistic Studies, Vol. 26 (2002) (in Chinese)
12. He, L., Chu, M., Lu, S.N., Qian, Y., Feng, Y.Q.: Studies on Prosodic Hierarchy Annotation in Mandarin Speech Corpus, proc. of the 5th National Conference on Phonetics, (2001) 323-326 (in Chinese)
13. Qian, Y., Chu, M., Pan, W.Y.: The Acoustic Cues of Boundaries of Prosodic Hierarchy in Mandarin, proc. of 5th National Conference on Phonetics, (2001) 70-74 (in Chinese)
14. Qian, Y., Chu, M. and Peng, H.: Segmenting Unrestricted Chinese Text into Prosodic Words Instead of Lexical Words, proc. of ICASSP2001, Salt Lake City (2001)
15. Chu, M., Wang, Y.J., Bao, M.Z.: Local Syntactic Constraints and Length Constraints in the Rhythmic Organization of Chinese Putonghua, Collection of Linguistic Studies, Vol. 30 (2005) 129-146 (in Chinese)
16. Chu, M, Peng, H., Zhao, Y., Niu, Z, Chang, E.: Microsoft Mulan — a Bilingual TTS Systems, proc. of ICASSP2003, Hong Kong (2003)

Prosodic Boundary Prediction Based on Maximum Entropy Model with Error-Driven Modification*

Xiaonan Zhang, Jun Xu, and Lianhong Cai

Key Laboratory of Pervasive Computing (Tsinghua University), Ministry of Education,
Beijing 100084
zxndelia724@gmail.com, xujun00@mails.tsinghua.edu.cn,
clh-dcs@tsinghua.edu.cn

Abstract. Prosodic boundary prediction is the key to improving the intelligibility and naturalness of synthetic speech for a TTS system. This paper investigated the problem of automatic segmentation of prosodic word and prosodic phrase, which are two fundamental layers in the hierarchical prosodic structure of Mandarin Chinese. Maximum Entropy (ME) Model was used at the front end for both prosodic word and prosodic phrase prediction, but with different feature selection schemes. A multi-pass prediction approach was adopted. Besides, an error-driven rule-based modification module was introduced into the back end to amend the initial prediction. Experiments showed that this combined approach outperformed many other methods like C4.5 and TBL.

Keywords: Prosodic Boundary Prediction, Maximum Entropy Model, Error-Driven Rule-Based Modification.

1 Introduction

When people talk, they rarely speak out a whole sentence without a break. Instead, an utterance is divided into smaller units with perceivable boundaries between them. These phonetic spurts or chunks of speech, which are commonly known as prosodic units, signal the internal structure of the message and serve important function in helping people to clearly express their ideas as well as their feelings.

These units are different in their categories, levels, and boundary strength. Smaller and lower-level units are contained in larger and higher-level units to form a prosodic hierarchy. In Mandarin Chinese, this hierarchical structure is often simplified to 3 layers [1] (from down to top): prosodic word, prosodic phrase and intonation phrase.

In current TTS, input text is firstly processed by a Text Analysis Model, whose output is a string of syntactic words, each with a POS tagging. Then its prosodic structure is expected to be constructed out of linguistic information to enhance the naturalness and understandability of the synthetic speech. However, as grammatical structure does not necessarily correspond to its prosodic counterpart, misjudgments

* This work is supported by China National Natural Science Foundation(60433030, 60418012).

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 149 – 160, 2006.

always occur in assigning the proper prosodic boundary, which has become the major impediment for TTS systems to achieve human-like performance.

That’s why more and more attention has been paid to addressing the problem of automatic prosodic boundary prediction. For Mandarin, as intonation phrases are usually distinguished by punctuation marks, most efforts focus on locating prosodic word and prosodic phrase boundaries based on syntactic information.

In the earlier time, rule-based methods were usually adopted. It mainly starts from Gee and Grosjean's work on performance structures [2], and has had various extensions over the years. For Chinese, similar investigation has also been carried out into the formation of prosodic constitutes, such as that reported by Jianfen Cao [1][3] and Hongjun Wang [4]. The central idea of all these work is to find some explicit rules that could recreate the prosodic structure of a sentence from syntax, by way of a large number of experiments and empirical observation. This method is easily explicable and understandable, but also poses strict demand for the system developer to summarize these rules. Moreover, it is hard to update and improve, and the set of rules is usually constrained to one branch of language, which hinders its general application.

With the availability of increasing prosodically annotated corpora and the rapid development of statistical learning, stochastic-based approach has been more and more widely used in prosodic boundary prediction. As in most cases, it is assumed that syntactic word is the smallest unit (i.e. leaf node) in a prosodic hierarchy tree, the task of building prosodic structure could be reduced to deciding the type for each syntactic word boundary, which is actually a classification problem. Thus many different statistical methods used for classification have been tried, such as Classification and Regression Tree (CART) used by Wang and Hirschberg [5], and Hidden Markov Model proposed by Paul and Alan [6]. Researchers in Chinese have also begun to adopt this approach during recent years. Besides those mentioned above, Zhao has described methods for automatically predicting prosodic phrase by combining decision tree and TBL [7]. And in Li’s experiment, he attempted to predict prosody phrase break based on Maximum Entropy Model [8]. Generally, these methods relate each boundary site with some features (e.g. length and POS of adjacent words). By extracting and absorbing these features from a large collection of annotated sentences, a statistical model is trained and then applied to unlabeled texts. For each potential boundary site in the text, a probability is estimated for each possible outcome, and the one with largest likelihood is determined as the correct type.

In this paper, we proposed to predict the prosodic boundary using Maximum Entropy Model, which nowadays has gained more and more popularity with NLP. Unlike previous efforts, we applied it to both prosodic word and prosodic phrase boundary labeling, and a multi-pass approach was employed for the latter task. Moreover, an error-driven rule-based modification module was added at the back end to improve the performance furthermore.

The rest of this paper is organized as following. Section 2 first introduced the Maximum Entropy Model briefly. Then the feature selection method and the multi-pass procedure for prosodic boundary prediction are presented. Section 3 described the back-end modification module. Experiments and results are given in Section 4. Section 5 gives conclusions.

2 Prosodic Boudary Prediction Based on Maximum Entropy Model

In our experiment, a basic assumption is that a prosodic boundary only occurs at syntactic word boundaries. For Mandarin Chinese, it is reasonable as statistics show that only 6% of prosodic words are part of a long syntactic word, and all the rest agree with this assumption.

Given a string of consecutive syntactic words, for each boundary between two of them (say w_i and w_{i+1}), there are 3 types: LW (w_i and $w_{i+1}$are within the same prosodic word), PW(within the same prosodic phase but different prosodic words) and PPh(within different prosodic phrases). Then our task comes down to deciding the right type for each syntactic word boundary, which could be accomplished with a Maximum Entropy Model.

2.1 Maximum Entropy Modeling

Consider a random process that produces an output value *y* based on some contextual information *x*, with *x* and *y* being a member of a finite set X and Y respectively. In our case, *y* is the type of a syntactic word boundary (i.e. LW, PW or PPh), and *x* could include any available information about that boundary.

Our task is to construct a stochastic model that accurately represents the behavior of the random process. In other words, it should give a reliable estimation of *p(ylx)*, which denotes the conditional probability that, given a context x, the process will output y.

For this purpose, we observe the behavior of the random process for some time, collecting N samples $(x_1, y_1), (x_2, y_2) \ldots\ldots (x_N, y_N)$. To express these facts, a **feature function** or **feature** for short is defined as:

$$fi(x, y) = \begin{cases} 1 & \text{if } y = y_i \text{ and } x = x_i \\ 0 & \text{Otherwise} \end{cases}$$

The expected value of each feature f_i with respect to the statistics of training samples could then be calculated as:

$$\bar{p}(f_i) = \sum_{(x,y)} \bar{p}(x, y) f_i(x, y) \ , \tag{1}$$

where $\bar{p}(x, y)$ is the empirical probability distribution of the samples, defined by:

$$\bar{p}(x, y) \equiv \frac{1}{N} \times \text{number of times that } (x, y) \text{ occurs in the sample} \ . \tag{2}$$

On the other hand, the expected value of f_i with respect to the unknown model *p(ylx)* is:

$$p(f_i) = \sum_{(x,y)} \overline{p}(x)p(y|x)f_i(x,y) \ , \tag{3}$$

where $\overline{p}(x)$ is the empirical distribution of x in the training sample. We require the model to accord with the observed statistics by constraining this value to be the same as the expected value of f in the training set. That is, for each f_i

$$p(f_i) = \overline{p}(f_i) \ . \tag{4}$$

Requirement (4) is called a **constraint equation** or simply a **constraint.**

Combining (1), (3) and (4) we have:

$$\sum_{(x,y)} \overline{p}(x)\, p(y \mid x)\, f_i(x,y) = \sum_{(x,y)} \overline{p}(x,y)\, f_i(x,y) \ . \tag{5}$$

Suppose we have n features, then all the probability distribution that satisfy the constraints exerted by these features constitute a set C:

$$C \equiv \left\{ p(y \mid x) \mid p(f_i) = \overline{p}(f_i) \quad for \ \ i \in \{1,2,...,n\} \right\} \ . \tag{6}$$

Among all the models p in C, the maximum entropy philosophy dictates that we select the one with maximum conditional entropy [9]

$$H(p) \equiv -\sum_{x,y} \overline{p}(x)\, p(y \mid x) \log p(y \mid x) \ , \tag{7}$$

and

$$p^* = \arg\max_{p \in C} H(p) \ . \tag{8}$$

It is a constrained optimization problem to find p^*. The target maximum entropy model has the following form[9]:

$$p^*(y \mid x) = \frac{1}{Z_\lambda(x)} \exp(\sum_i \lambda_i f_i(x,y)) \ . \tag{9}$$

where $Z_\lambda(x)$ is a normalizing constant and λ_i is a Lagrange multiplier which is commonly computed from the training set using GIS algorithm. Detailed steps are omitted here.

2.2 Feature Selection Strategy

The principle of Maximum Entropy Model is to agree with all that is known and assume nothing about what is unknown. Yet it poses another important question: how to find appropriate facts that are most relevant to the task in hand? Put another way, how to select a limited number of features that represents the 'known' fully and accurately?

In the first place, as prosodic phrase lies above prosodic word in the prosodic hierarchy, it should exhibit some 'higher-level' features than the latter. Taking this into account, we built two distinct models by incorporating into them different features for prosodic word and prosodic phrase prediction respectively.

Like Li [8], we used a semi-automatic approach for feature selection. First, feature 'templates' are manually designed, which in effect defined the space of candidate features; then the most "useful" features are selected automatically using a simple count cut-off method with a threshold of 3.

The feature 'templates' are so devised as to capture as much information about the random process as possible. For our specific application, most commonly used features include POS (Part of Speech Tagging), WLen (length in syllables) and Word (the word itself) of the words surrounding the boundary, which have also proved to be the most important determinants of prosodic boundary types [10]. On account of this, we added them into too both templates, with a window length of 2 for POS, i.e. we considered the POS of 2 words immediately before and after the boundary in question, and a window length of 1 for WLen and Word. A point to note, though, is that 'word' has different meaning under the two scenarios. For prosodic word, it indicates syntactic word and is readily available from the input text. For prosodic phrase, which is built upon prosodic words rather than syntactic words, the meaning accordingly changes to prosodic word. Here 'POS' property of a prosodic word is acquired by simply concatenating POS's of the syntactic words it contains (e.g. POS of '我们/rr 的/ud' is 'rr ud')

Besides these widely used features, another category of features—'dynamic feature' were also introduced into the templates. The first is 'lastType', which denotes the last prosodic boundary type. The motive for adding this information came from the observation that current boundary type is influenced by that of last one, which applies to prosodic word as well as prosodic phrase boundary. For example, a 'lastType' of PPh could well reduce the possibility that current boundary is still PPh.

The second was specially proposed for prosodic phrase segmentation. We noted that, to a large extent, insertion of prosodic phrase boundaries in natural spoken language is to balance the length of the constituents in the output. Hence it is not surprising that most PPh breaks occur in the middle part of a long sentence, and a prosodic phrase is usually 5~7 syllables long, but rarely shorter than 3 or longer than 9 syllables. For this reason, we took into consideration length measures by including 'dBack' and 'dFront' in our templates for prosodic phrase prediction, which means the distance (in syllables) from current boundary to the last and next nearest PPh location.

This category of features is by definition 'dynamic' in that they rely on the result of previous prediction, and remains unknown until judgment on preceding boundaries has been made. By contrast, the usual 'static' features are fixed and known all the way once the input is given.

The feature templates contained both atomic and composed ones. Atomic templates considered only one element mentioned above, while composed templates are combination of atomic ones. Table 1 lists all the atomic templates.

Table 1. Atomic Templates Used in Two Maximum Entropy Model

	PW&PPh Prediction 1st Pass (Model 1)		PPh Prediction 2nd&3rd Pass(Model 2)	
	Symbol	**Meaning**	**Symbol**	**Meaning**
Atomic Templates	POS-2 POS-1 POS+1 POS+2	POS of the 1st/2nd syntactic word before/after the boundary	POS-2 POS-1 POS+1 POS+2	POS of the 1st/2nd prosodic word before/after the boundary
	Word-1 Word+1	1st syntactic word itself before/after the boundary	Word-1 Word+1	1st prosodic word itself before/after the boundary
	WLen-1 WLen+1	length of 1st syntactic word before/after the boundary	WLen-1 WLen+1	length of 1st prosodic word before/after the boundary
	lastType	boundary type after last syntactic word (LW/PW/PPh)	lastType	boundary type after last prosodic word (PW/PPh)
			dFront/ dBack	distance from current position to last/next PPh boundary

2.3 Multi-pass Prediction

In both our experience and experiments, we found that it's much easier to locate prosodic word boundaries accurately. It could be explained by the observation that distribution of this kind of boundaries largely depends on local syntactic constraints and exhibits more regular patterns that could be derived from low-level syntax analysis. On the other hand, prosodic phrasing is a compromise between the need to respect the syntax structure of the sentence and the prosodic constraints, which could hardly be decided in the normal one-pass classification solution.

That's why we came up with the idea of multi-pass prediction to determine prosodic phrase boundaries. The whole process is described in Figure 1. As mentioned earlier, 2 separate models were trained with different feature sets during the training stage. In testing, the 1st-pass prediction used Model 1 at every syntactic word boundary to decide its type: LW, PW, or PPh. At this time, our major concern was to differentiate between PW and LW boundaries, and merely those most 'credible' PPh's were labeled as PPh's. That is, only when Model 1 decided that the probability of a boundary to be PPh is higher than a certain threshold (say threshold1), were we assured that it actually was PPh. Otherwise we still classified it as PW and left it to the next pass. It is worth to mention that though Model 1 was mainly targeted at PW prediction, it's sensible and necessary to label out some PPh's at the same time. For one thing, those PPh's with a high degree of confidence are mostly where we 'have to break' governed by syntax or grammatical constraints. For another, identifying these PPh positions also enabled us to acquire the 'dBack' feature in following predictions.

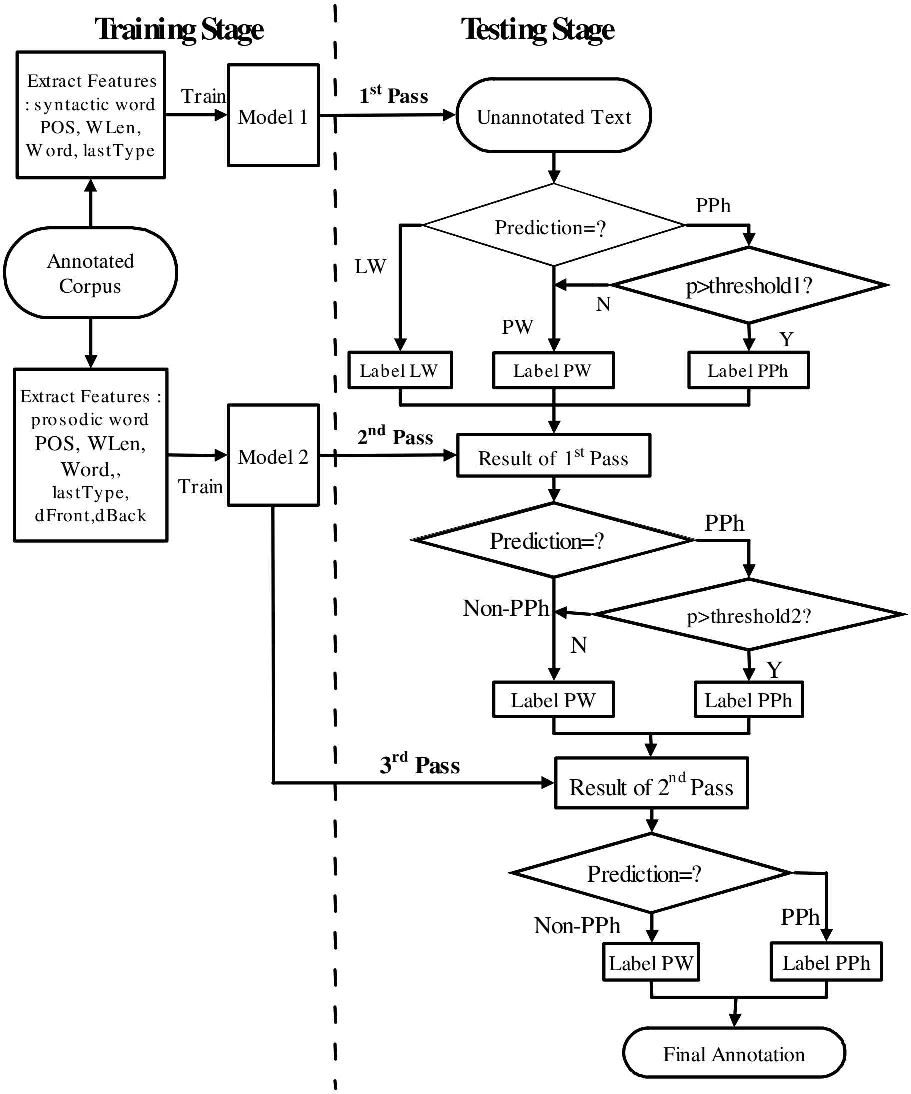


Fig. 1. Procedure of Multi-Pass Prediction

The 2nd and 3rd pass only worked on PW boundaries labeled in the 1st pass. They both used Model 2 to decide whether a PW indeed was PW, or should be classified as PPh. The only difference with these two pass is that during the 2nd pass, we still didn't take the result literally: only when the estimated probability of a boundary to be PPh was higher than threshold2, did we trust it to be an 'authentic' PPh. However, in the last pass, we accepted the model's judgment unconditionally.

We did this mainly because those PPh's decided in the latter stage of our prediction chiefly correspond to those breaks that we 'don't have to make but could make' out of prosodic constraints, and thus had better to be refined step by step to achieve the best balance in length.

Another question unaddressed is the set of threshold1 and threshold2, which turned out to have a considerable influence on the final outcome. After repeated experiments, it was found that a value of 0.65 and 0.7 for threshold1 and threshold2 respectively achieved the best performance.

3 Error-Driven Rule-Based Modification

In our preliminary experiment using only Maximum Model for prediction, we found that there were always some obvious mistakes that humans would never commit as they obviously contradicted to some ‘fixed patterns’ we were accustomed to. It then occurred to us that these mistakes might be corrected with manually-made rules.

3.1 Rules

Every rule is a pair with the form of ‘predicate => action’. When and only when the pre-condition described by ‘predicate’ is satisfied, will a rule be activated, and then corresponding ‘action’ will be taken.

For example, the fact that ‘A boundary succeeded by syntactic word “的”or “得”must be a LW boundary’ could be written as the following rule:

$$\mathit{WORD}\text{-}1 = \text{的}\ \mathit{or}\ \mathit{WORD}\text{-}1 = \text{得} \Rightarrow \mathit{Boudary} \leftarrow \mathit{LW}$$

3.2 Basic Process

The rule-based modification module was added at the back-end of the system to amend the prediction from Maximum Entropy Model. To evaluate whether adding a rule does improve the performance, the metric ‘F-Score’ (detailed later in 4.1) was used. A brief working process is shown in Figure 2.

We compared the result of automatic annotation with manual annotation to detect errors made by the machine. By observing the statistics, a rule was derived to correct them. In most cases, errors first got rectified were those most amendable ones, i.e., errors which exhibited some evident patterns. Every time a rule was worked out, it was tried out on the whole testing corpus to see whether the resulting new F-Score was notably higher than that of last time. If it didn’t, the rule was just ignored; otherwise it was adopted and applied. New errors might occur and this process repeated, until no rules could be manually found.

The underlying idea of this module is a bit like that of TBL (Transformation-Based Error-Driven Learning) [11]: It starts from an initial state, and by use of a series of transformation rules, it modifies the result bit by bit to achieve the best score according to the objective function used. It’s only that in TBL, the transformations are learnt automatically (typically by greedy search algorithms); but in our solution, these rules are manually formulated to avoid the heavy computational cost.

For now a total of 15 rules were added.

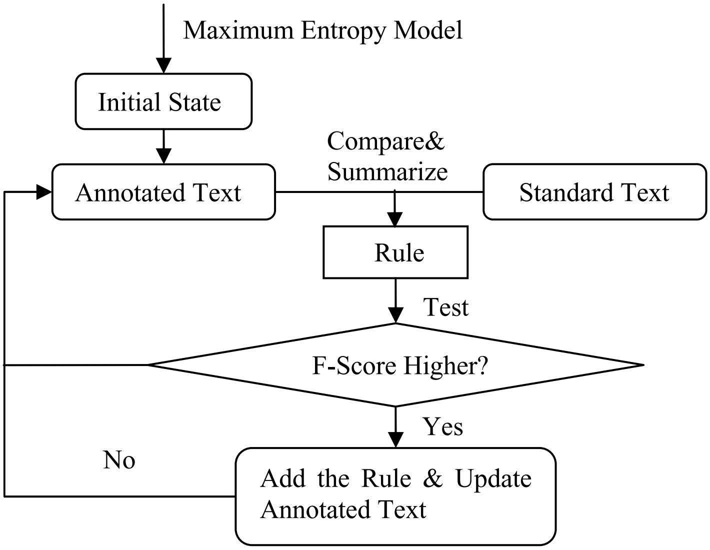


Fig. 2. Basic Process of Post-modification

4 Experiment and Result

4.1 Preparation

Our raw corpus comprises 10000 sentences randomly selected from People's Daily 2000. Each sentence had been segmented into a sequence of syntactic words with POS tags according to "Specification for Corpus Processing at Peking University: Word Segmentation, POS Tagging and Phonetic Notation" (shortened as "Specifiation 2003"). On average, one sentence contains 45.24 syllables and 25.15 syntactic words.

Then this corpus was prosodically annotated by two trained people, who were consistent on more than 90% of their annotation. Both prosodic words and prosodic phrase boundaries were marked out. The whole process was guided and supervised by an expert (Jianfen Cao of Chinese Academy of Social Sciences).

Among the 10000 sentences, 4500 were used for training and 2000 were used for testing in all of the following experiments. The two sets did not overlap each other.

4.2 Evaluation Criteria

Since subjective tests are time-consuming and costly to perform, we adopted an objective point of reference.

As mentioned earlier, there are altogether 3 prosodic boundary types between two syntactic words w_i and w_{i+1}: LW, PW and PPh. For simplicity in notation, we here refer to them as B_0, B_1 and B_2 respectively. To evaluate the performance of our system, the prosodic boundaries automatically assigned to the testing set were compared to human-annotation. In this way a confusion matrix was acquired, as shown in Table 2. In the table, C_{ij} denotes the number of boundaries manually labeled as B_i and predicted to be B_j.

Table 2. Confustion Maxtrix

Manually Labeled Type	**Predicted Type**		
	B_0	B_1	B_2
B_0	C_{00}	C_{01}	C_{02}
B_1	C_{10}	C_{11}	C_{12}
B_2	C_{20}	C_{21}	C_{22}

Our evaluation metric *Precision* and *Recall* were computed as:
For prosodic word boundary prediction:

$$\text{Precision}_1 = \frac{\sum_{j=1}^{2}\sum_{i=1}^{2} C_{ij}}{\sum_{j=0}^{2}\sum_{i=1}^{2} C_{ij}} , \quad \text{Recall}_1 = \frac{\sum_{j=1}^{2}\sum_{i=1}^{2} C_{ij}}{\sum_{i=0}^{2}\sum_{j=1}^{2} C_{ij}} . \tag{10}$$

For prosodic phrase boudnary prediction:

$$\text{Precision}_2 = \frac{C_{22}}{\sum_{j=0}^{2} C_{2j}} , \quad \text{Recall}_2 = \frac{C_{22}}{\sum_{i=0}^{2} C_{i2}} . \tag{11}$$

Another measurement F-Score takes both into consideration:

$$F\text{-Score}_i = \frac{2\times \text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \quad (i = 1, 2) . \tag{12}$$

4.3 Effect of Adding Rules

Figure 3 shows the test results for prosodic words segmentation when the number of rules gradually increased from 0 to 15. Since Maximum Entropy Model alone was able to achieve a relatively high accuracy with a large training corpus (4500 sentences in our case), the post-processing module doesn't seem to be playing a significant part. Yet there is still notable rise in all three measures. When training material is of a small size and linguistic feature values are sparse, more remarkable improvement could be expected.

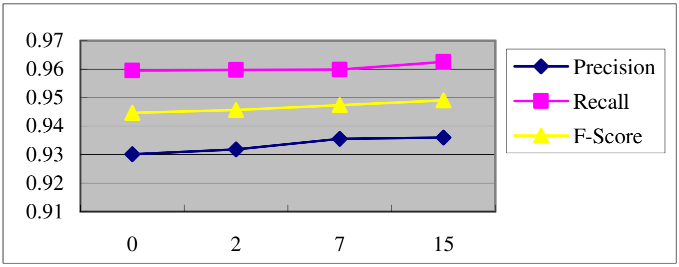


Fig. 3. Prosodic Word Boundary Prediction Result When Adding Rules

4.4 General Performance

Table 4 gives the testing result of our system, with threshold1=0.65 and threshold2= 0.7 for multi-pass prediction. Best results of some other approaches adopted in previous experiments are also listed for comparison.

Table 4. Best Test Result in Comparison with Other Methods

	Prosodic Word			**Prosodic Phrase**		
	Precision	Recall	F-Score	Precision	Recall	F-Score
C4.5 [7]	0.814	0.822	0.818	0.712	0.829	0.766
TBL [7]	0.782	0.848	0.814	0.853	0.613	0.713
ME Model[8]	N/A	N/A	N/A	N/A	N/A	0.652
Our Approach	0.936	0.963	**0.949**	0.798	0.784	**0.791**

Due to difference in the corpora and evaluation metric, these results may not be comparable in all respects. Yet from the statistics above, we could safely say that our approach is a successful attempt towards prosodic boudary prediction.

5 Conclusions

In this paper, we addressed the problem of prosodic boudary prediction based on syntactic information. The Maximum Entropy Model was utilized with two separate instantiation for prosodic word and prosodic phrase segmentation. Our feature selection strategy was distinctive in that it not only drew on generally 'static' syntactic context, but also considered the interplay among successive boundary positions by incorporating 'dynamic' features into the model. It gave a satisfying performance especially for prosodic word prediction. For prosodic phrase prediction, even though lack of high-level syntactic and semantic information impeded its accurate prediction, a multi-pass procedure served to strike a better prosodic balance. Besides, the combination of machine learning power and human wisdom through error-driven ruled-based modification further enhanced its performance.

Future work should focus on extraction of ‘deeper’ contextual information such as sense group and semantic chunk to aid the perception of prosodic phrase boundaries. Moreover, the inherent uncertainty in prosody structure in natural language may require a more flexible approach to its prediction, possibly by using a minimum error-rate criterion (MERC) [12] in place of the traditional maximum correct-rate criterion currently adopted by us.

References

[1] Jianfen Cao: Prediction of Prosodic Organization Based on Grammatical Information. Journal of Chinese Information Processing, Vol. 17. (2003) 41–46

[2] Gee J.P., Grosjean F: Performance structures: A psycholinguistic and Linguistic Appraisal. Cognitive Psychology, Vol. 15. (1983) 411–458

[3] Jianfen Cao, Weibin Zhu: Syntactic and Lexical Constraint in Prosodic Segmentation and Grouping. In: Proceedings of Speech Prosody 2002. Aix-en-Provence, France (2002)

[4] Hongjun Wang: Prosodic words and prosodic phrases in Chinese. Chinese Language,Vol. 6. (2000) 525–536

[5] Wang M., Hirschberg J.: Predicting Intonational Boundaries Automatically from Text, In: the ATIS Domain Proceedings of the DARPA Speech and Natural Language Workshop, (199l) 378–383

[6] Paul Taylor, Alan. W. Black: Assigning phrase breaks from part-of speech sequences, Computer Speech and Language, Vol. 12(4). (1998) 99–117

[7] Z Sheng, T Jianhua, C Lianhong: Learning rules for Chinese prosodic phrase prediction. International Conference on Computational Linguistics, Proceeding of the first SIGHAN workshop on Chinese language processing, Vol. 18. (2002)

[8] Jian-Feng Li, Guo-Ping Hu, Renhua Wang: Chinese prosody phrase break prediction based on maximum entropy model, In: Interspeech 2004, Jeju Island, Korea. (2004) 729–732

[9] BERGER AL, STEPHEN A, DELLA PIETRA SA, et al: A maximum entropy approach to natural language processing [J]. Computational Linguistics, Vol. 22(1). (1996) 39–71

[10] Min Zheng, Lianhong Cai: Prosodic Constituents Segmentation and Syntax of Chinese: In: Proceeding of 5th Chinese Lexical Semantics Workshop, Singapore (2004)

[11] Brill, Eric: Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. Computational Linguistics. Vol. 21(4). 543–565

[12] Min Chu: The Uncertainty in Prosody of Natural Speech and Its Application in Speech Synthesis, Journal of Chinese Information Processing, Vol. 18(4). (2004) 66–71

Prosodic Words Prediction from Lexicon Words with CRF and TBL Joint Method*

Heng Kang and Wenju Liu

National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Sciences
{hkang, lwj}@nlpr.ia.ac.cn

Abstract. Predicting prosodic words boundaries will directly influence the naturalness of synthetic speech, because prosodic word is at the lowest level of prosody hierarchy. In this paper, a Chinese prosodic phrasing method based on CRF and TBL model is proposed. First a CRF model is trained to predict the prosodic words boundaries from lexicon words. After that we apply a TBL based error driven learning approach to refine the results. The experiments shows that this joint method performs much better than HMM.

Keywords: prosodic words, lexicon words, CRF, TBL.

1 Introduction

In Chinese spoken language, the prosodic structure of an utterance can be viewed as three levels: prosodic word, prosodic phrase and intonation phrase[1]. The prosodic word (P-word) is defined as a group of syllables that are uttered closely and continuously in speech[2]. No boundary should be perceived within a prosodic word. That is to say, prosodic word is the basic prosodic structure in spoken Chinese. And in speech synthesis, experiments show the TTS system using the prosodic words as the basic prosodic units can get high naturalness than using lexicon words (L-word)[3]. Because the prosodic words influence the rhythm of synthetic utterances very much, it is of much importance to predict prosodic words from the input text.

Although lexicon words segmentation technology has become mature in Chinese language processing, it is still a difficult work on prediction prosodic words from unrestricted text. In all of previous work, Hidden Markov Model (HMM) based methods are adopted in many TTS systems because of the elegant methodology. Although most of them have taken full use of the feature information, the segmentation results are not good enough. The reason lies in the structure of HMM[4]. HMM is a probabilistic model of the way in which the data and labels are generated. This model

* This work is supported in part by the China National Nature Science Foundation (No. 60172055, No. 60121302), the Beijing Nature Science Foundation (No.4042025) and the National Fundamental Research Program (973) of China (2004CB318105).

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 161 – 168, 2006.

has some drawbacks. Firstly the structure of HMM is often a poor model of the true process to produce data. Because of its first-order Markov property, any relationship between two labels must communicates via the intervening status, which cannot in general capture the relationships. Secondly is HMM generates each datum only from the corresponding status, which makes it difficult to utilize an input sliding window.

In this paper we proposed a prosodic words prediction method based on Conditional Random Field(CRF)[5] and Transformation Based Learning(TBL)[6,7] joint model. Firstly a CRF model is trained to predict the prosodic words boundaries from lexicon words. After that we apply a TBL based error driven learning approach to refine the results. The experiments shows that this joint method performs much better than HMM. In our project we apply this model on both prosodic words grouping and splitting.

This paper is organized as follows. Section 2 gives an introduction to the prosodic words. Section 3 describes our CRF model based method to predict prosodic words. Section 4 gives the description of the TBL method to refine the CRF predicting results. Experiments results will be presented in section 5. Finally we conclude our paper.

2 Prosodic Words

2.1 Prosodic Words and Lexicon Words

In Chinese, the hierarchy of prosody is not identical to that of syntax. The prosodic word is defined as a group of syllables that are uttered closely and continuously. While lexicon words is according to a lexicon. For example, in a Chinese sentence "他的帽子太大了(His hat is too large)" can be segmented to lexicon words "他/的/帽子/太/大/了". But in spoken speech the utterance should be segmented to prosodic words "他的/帽子/太大了". It can be seen that there is not a direct mapping between lexicon words and prosodic words. Statistical results show that only about half of lexicon words are prosodic words as well.

The prosodic words cannot be directly stored in a lexicon, because they will change greatly in different context. Therefore we cannot simply lookup in a lexicon to segment a sentence into prosodic words.

Studies show that most P-words consist of two characters, and very few P-words consist of 3 characters or above. This is due to the bi-character rhythm demand to build the prosodic foots in the phonology. Because of this reason, some one-character lexicon words should be grouped into one P-word, and long L-word is tended to be split into several P-words.

Our research is based on a corpus with 13000 sentences. For each sentence the prosodic words boundaries are manually labeled. The statistical results show that there are about 110,000 lexicon words total, and 81,000 prosodic words. Figure 1 illustrates the length distribution of P-words and L-words in this corpus. From this figure we can find that most P-words have 2 Chinese characters.

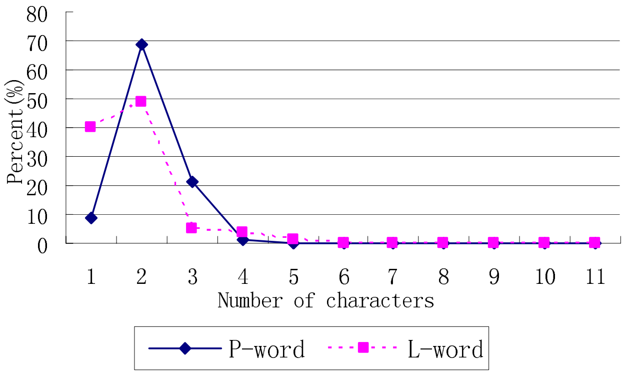


Fig. 1. Length distribution of P-words and L-words in the corpus

2.2 Prosodic Words Labeling

In order to predict prosodic words from lexicon words automatically, the training corpus should be labeled manually by a set of guidelines[2]. The labels include lexicon words boundaries, the POS(part of speech) of each lexicon words, and the prosodic words boundaries. To get high consistent labeling, only one annotator is asked to do this labeling work in our experiments.

We take the Chinese sentence “在这幸福的日子里我们歌唱祖国。” for example:

(1) 在这幸福的日子里我们歌唱祖国。
(2) 在/p 这/r 幸福/a 的/u 日子/n 里/f 我们/r 歌唱/v 祖国/n 。/w
(3) 在这| 幸福的| 日子里| 我们| 歌唱| 祖国。
(4) 在/p 这/r | 幸福/a 的/u | 日子/n 里/f | 我们/r | 歌唱/v | 祖国/n 。/w

where (1) is the original text. After POS tagging (2) is gotten, in which ‘/’ means the boundaries and the symbols followed by ‘/’ is the POS tag. (3) is labeled with the prosodic words, in which ‘|’ means the prosodic words boundaries. For better usage of the boundaries and POS tag, (2) and (3) are combined to (4).

All the 13000 sentences in our corpus are labeled like this.

3 CRF Based Method to Predict Prosodic Words

HMM is an elegant and easy methodology that is adopted in many TTS system. It is a probabilistic model in which data and status are generated. However it suffer from some drawbacks. Firstly the structure of HMM is often a poor model of the true process to produce data. Because of its first-order Markov property, any relationship between two labels must communicates via the intervening status, which cannot in general capture the relationships. Secondly is HMM generates each datum only from the corresponding status, which makes it difficult to utilize an input sliding window.

Maximum Entropy Markov Models (MEMM)[8] attempt to maximize the conditional likelihood of data via a maximum entropy method. Although this model supports long-distance interactions, unfortunately it suffer from a label bias problem.

3.1 Introduction to CRF model

Conditional Random Fields are introduced to overcome these problems. CRFs are undirected graphical models that encode a conditional probability distribution with a given set of features. In the special case in which the designated output nodes of the graphical model are linked by edges in a linear chain, CRFs make a first-order Markov independence assumption among output nodes. Fig.2 shows the graphical structure of a chain-structured CRFs.

For sequential data $X = x_1 ... x_T$ and their corresponding labels (status) $Y = y_1 ... y_T$, a linear chain structure CRF defines the conditional probability as

$$P_\Lambda(Y \mid X) = \frac{1}{Z_X} \exp(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_{t-1}, y_t, x, t)) \tag{1}$$

where Z$_x$ is the per-input normalization that makes the probability of all state sequences sum to one; $f_k(y_{t-1}, y_t, x, t)$ is a feature function which is often binary-valued, but can be real-valued, and λ_k is a learnt weight associated with feature f_k . The feature functions can measure any aspect of a state transition y and the entire observation sequence x centered at the current time step t. Large positive values for λ_k indicate a preference for such an event; large negative values make the event unlikely.

The model parameters f_k can be estimated by maximum likelihood—maximizing the conditional probability of a set of label sequences, each given their corresponding input sequences. The log-likelihood of the training set is

$$\begin{aligned} L_\Lambda &= \sum_i \log P_\Lambda(y_i \mid x_i) \\ &= \sum_i \left(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_{t-1}, y_t, x, t) - \log Z_{xi} \right) \end{aligned} \tag{2}$$

Traditional maximum entropy learning algorithms, such as GIS and IIS[9] can be used to train CRFs.

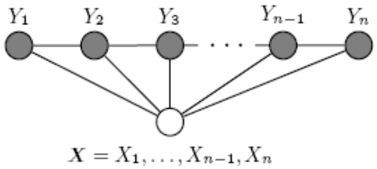


Fig. 2. Graphical structure of a chain-structured CRFs for sequences

For the given observation sequential data, the most probable label sequence can be determined by

$$y^* = \arg\max_{y} P_{\Lambda}(Y \mid X) \tag{3}$$

which can be efficiently determined using the Viterbi algorithm[10]. An N-best list of labeling sequences can also be obtained using modified Viterbi algorithm and A* search[11].

3.2 CRF for Prosodic Words Prediction

For automatically processing the labels by computers, the manually labeled data should be formatted as follows:

The sentence “在/p 这/r | 幸福/a 的/u | 日子/n 里/f | 我们/r | 歌唱/v | 祖国/n 。/w “ is formatted to “在/p/B 这/r/E 幸福/a/B 的/u/E 日子/n/B 里/f/E 我们/r/S 歌唱/v/S 祖国/n/S 。/w/W”, in which ‘/B’ represents this lexicon is at the beginning of a prosodic word, ‘/E’ means this lexicon word is at the end of a prosodic word, and ‘/I’ means it is at the intermediate part.

After the labeling we can find that prosodic words prediction is a typical tagging problem which can be described as: given the observation sequence $X = x_1 ... x_T$, determine the corresponding labels $Y = y_1 y_2 ... y_N$. This can be solved directly by CRF.

To utilize the flexibility of CRF and considering the prosodic words prediction problem, we use the features in table 1.

In addition, the word length of the current prosodic words is used as the feature. For the lexicon word “幸福/a” in the last example, the feature for length of the prosodic words is f(PW-3B) = 1, which means this lexicon is at the beginning of a 3-characters-long prosodic words. Because most prosodic words have 2 or 3 Chinese characters, this type of feature is very import.

Table 1. Features used in CRF modeling

Features	Explanations
LW_{-2}	second previous lexicon word
LW_{-1}	previous lexicon word
LW_0	current lexicon word
LW_1	next lexicon word
LW_2	second next lexicon word
LW_0LW_1	Current lexicon word and next lexicon word
$LW_{-1}LW_0$	previous lexicon word and current lexicon word
$LW_{-2}LW_{-1}$	second previous lexicon word and previous lexicon word
$LW_{-1}LW_0LW_1$	previous lexicon word, current lexicon word and next lexicon word

4 Refining the Result with TBL

Although we use a very large corpus to train the CRF model, the sparseness problem still occurs because of the statistical method. For this reason we try to make use of TBL[6,7] to refine the prediction results.

TBL is an algorithm that can automatically get rules from a set of templates. Comparison with statistical methods like CRF, TBL is not so sensitive to the data sparseness.

Fig. 3 illustrates how to select rules with TBL. Before learning stage, the program compares the initial labels and the mannul labels results sentence by sentence. If they are totally the same the sentence will be skipped, otherwise, a candidate rule is generated from a template.

In the learning stage, the evaluation process will apply each rule in the candidate rules set. A score is given according to the number of errors that the rule can amend. The rule with the highest score is recorded, and the amended results by it will be saved as the initial status of next loop. This evaluation-application loop runs until no more errors can be corrected.

For our application, the configuration is as follows:

(1) initial labels: results from CRF prediction

(2) rule templates: templates should be designed to consider the prosodic words problem. In our application, the templates are very like the features used in CRF model:

If (LW_{-1}:POS=P1 & LW_{-1}: POS=P2 & LW_{-1}:LENGTH=L) then PTAG1 =>PTAG2

which means, if the POS of previous lexicon word is P1 and POS of current lexicon word is P2 and length of previous lexicon word is L, then the tag PTAG1 is corrected to PTAG2. PTAG is in the set {B, I, E, S, W}.

(3) evaluation: a score is given according to the number of errors that the rule can amend. A rule with the highest score is selected. A threshold should be set that only when the score is more than the threshold the rule could be recorded.

In our experiment,

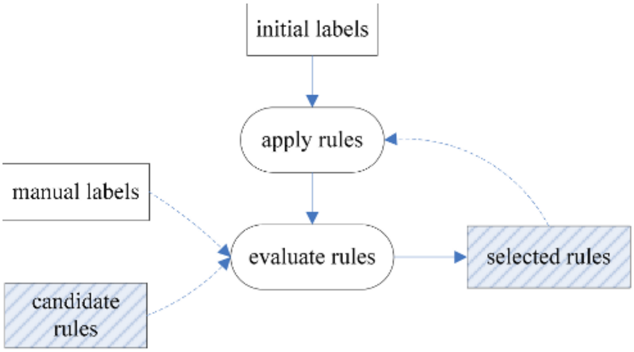


Fig. 3. TBL learning process

5 Experiments

For the Chinese prosody research, we collected and designed a large phonetically and prosodic enriched text corpus from different domains. In this large corpus there are about 13000 text sentences, which are labeled carefully by a well-trained researcher. We select 10000 sentences to train our model and the rest of them are for test. We also adopt HMM model based method for comparison.

In the training set, there are 120121 lexicon words and 90312 prosodic words. The longest lexicon word has 11 Chinese characters, and the longest prosodic word has 4 Chinese characters.

In our experiments, after training, we test on both training set and the test set. There are 2 evaluation criteria: precision and recall rate, which are defined as follows.

$$F_{pre} = \frac{N_1}{N_2} \times 100\% \tag{4}$$

$$F_{rec} = \frac{N_1}{N_3} \times 100\% \tag{5}$$

Where N_1 is the number of prosodic word boundaries predicted correctly, N_2 is the total number of prosodic word boundaries predicted, and N_3 is the total number of real prosodic word boundaries in the test set.

Table 2. Statistical results of the experiments

results models		10000 sentences close set		3000 sentences open set	
		precision	recall rate	precision	recall rate
No Model		59.07%	**96.71%**	59.65%	**96.54%**
CRF	CRF	90.52%	95.72%	90.12%	92.29%
	CRF + TBL	**95.87%**	95.90%	**93.22%**	94.44%
HMM	HMM	83.90%	94.77%	84.33%	94.52%

Table 2 illustrates our experimental results, in which “no model” method means the lexicon words boundaries is labeled as prosodic words boundaries directly. Apparently this method will get highest recall-rate and lowest precision.

From this table we can draw a conclusion that CRF model based method get high precision than HMM model based method, both in close set (training set) and open set (test set). And after applying TBL refinement, the precision and recall-rate increase more.

And we can also find that the precision and recall-rate don’t decrease much in the open set test. This indicates that the our method is robust and generalized.

6 Conclusion

In this paper, a Chinese prosodic phrasing method based on CRF and TBL model is proposed. First a CRF model is trained to predict the prosodic words boundaries from lexicon words. After that we apply a TBL based approach to refine the results. The experimental results show that this joint method performs much better than HMM.

Reference

1. Jianfen Cao, Acoustic phonetic features in the rhythm of Mandarin, The 4th national conference on modern phonetics, 1997
2. Min Chu, Yao Qian, Locating boundaries for prosodic constituents in unrestricted Mandarin texts, Computational linguistics and Chinese language processing, vol.6, no.1, 2002, pp.1-22
3. Yao Qian, Min Chu, Segmenting unrestricted Chinese text into prosodic words instead of lexicon words, Proceedings of the 2001 International conference on acoustic, speech and signal processing, 2001, Salt Lake City
4. Thomas G. Dietterich, Machine learning for sequential data: a review,
5. John Lafferty, Andrew McCallum, Fernando Pereiram, Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, Proceedings of the Eighteenth International Conference on Machine Learning, 2001
6. E. Brill, A rule-based approach to prepositional phrase attachment disambiguation, Proc. 15th international conference on computational linguistics, 1994, pp1198-1204
7. E. Brill, Automatic grammar induction and parsing free text: a transformation-based approach. Proc. Of the ARPA human language technology workshop, Princeton, N.J. 1993
8. McCallum, A., Freitag, D. & Pereira, F. Maximum Entropy Markov Models for Information Extraction and Segmentation, Proc. ICML 2000
9. S. della Pietra, V. della Pietra, and J. Lafferty, Inducing Features Of Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4), 1995
10. L. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Readings in Speech Recognition*, pages 267–296, 1990
11. R. Schwartz and Y. Chow, The N-best Algorithm: An Efficient and Exact Procedure for Finding the N most Likely Sentence Hypotheses, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1990*

Prosodic Word Prediction Using a Maximum Entropy Approach*

Honghui Dong[1], Jianhua Tao[1], and Bo Xu[2]

[1] National Laboratory of Pattern Recognition
{hhdong, jhtao}@nlpr.ia.ac.cn
[2] High Technology Innovation Center,
Institute of Automation, Chinese Academy of Sciences
xubo@hitic.ia.ac.cn

Abstract. As the basic prosodic unit, the prosodic word influences the naturalness and the intelligibility greatly. Although the research shows that the lexicon word are greatly different from the prosodic word, the lexicon word still provides the important cues for the prosodic word forming. The rhythm constraint is another important factor for the prosodic word prediction. Some lexicon word length patterns trend to be combined together. Based on the mapping relationship and the difference between the lexicon words and the prosodic words, the process of the prosodic word prediction is divided into two parts, grouping the lexicon word to the prosodic word and splitting the lexicon word into prosodic words. This paper proposes a maximum entropy method to model these two parts, respectively. The experiment results show that this maximum entropy model is competent for the prosodic word prediction task. In the word grouping model, a feature selection algorithm is used to induce more efficient features for the model, which not only decrease the feature number greatly, but also improve the model performance at the same time. And, the splitting model can correctly detect the prosodic word boundary in the lexicon word. The f-score of the prosodic word boundary prediction reaches 95.55%.

1 Introduction

The prosodic word is the basic unit of the prosodic structure, which greatly influence the naturalness of the TTS system. Not every lexicon word can be directly read as a prosodic word in the utterance. They are very different between each other. The perception experiment[1] shows that the TTS system using the prosodic word as the basic unit has much higher intelligibility and naturalness than using the lexicon word directly. At the same time, the prosodic words can not be stored in the lexicon as a lexicon word, since the prosodic words have not a certain linguistic property and the prosodic words will change greatly in the different contexts. Therefore, prosodic word detection from the text became a necessary step in the prosodic analysis module of the TTS system. In previous

* This work is supported by National Natural Science Foundation of China(60575032).

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 169–178, 2006.

research works, many methods were introduced to predict the prosodic word. [1] use a statistical rule based method and a CART based method, which mainly use the part-of-speech(POS) and word length feature. Although the syntax feature is useful to improve the precision and recall, the parsing will cost much more time[1]. [2] use the word POS and length information as the features, and introduce a dynamic programming method to predict the prosodic words. [3] use extended features to predict the prosodic word with a CART method. The extended features include totally three kinds of features: the POS feature, the base phrase, and the syntactic feature. All these methods use the lexical information as an important feature to predict the prosodic word. In these works it is shown that the lexicon word is deeply relative to the prosodic word.

In this paper, we study the relationship between the lexicon word and the prosodic word, and try to find the cues to detect the prosodic word from the lexical information. Based on the analysis, the process of prosodic word prediction is divided into two parts: grouping the lexicon words to the prosodic word and splitting the lexicon word into the prosodic words. And, a maximum entropy approach is introduced to model these two processes. A feature selection method based on the likelihood gain is used to induce more efficient features. The layout of this paper is as the follows: the problem analysis for the prosodic word is presented in section 2. The maximum entropy model for the prosodic word prediction is presented in section 3. Section 4 details the experiments and result analysis. At the end, section 5 gives our conclusions.

2 Problem Statement

For the mandarin prosody research, we designed and collected a large phonetically and prosodically enriched Mandarin speech corpus, of which the prosody structure is labelled by a well-trained person. 6000 utterances between 7 and 25 syllables are included in this study. Among this corpus, 4000 sentences are used for analysis or training, 1000 sentences for developing and the rest 1000 sentences for testing.

In the analysis corpus, it is discovered there are only about 50% lexicon words to be the prosodic word directly. There is not a one-to-one mapping relationship between the lexicon word(L-word) and the prosodic word(P-word). Figure 1 gives the length distribution of the P-words and the L-words in the analysis corpus. It is shown that the P-word is more likely to be two syllable long and very few P-words will have more than 3 syllables. There are much more mono-syllable L-words than the mono-syllable P-words. This accounts for why the mono-syllable L-words trend to bundle together into a P-word in the corpus. It is due to the bi-character rhythm demand of building the prosodic word. It is consistent with the phonetics theory [4][5]. On the other hand, the L-word longer than 3 is always split into several shorter P-words, which are two syllables long. Therefore, there are much difference between the L-word and the P-word. On the other hand, the difference can also provide some useful cues for predicting the P-word from the L-word. Actually, based on the process forming the P-words from the

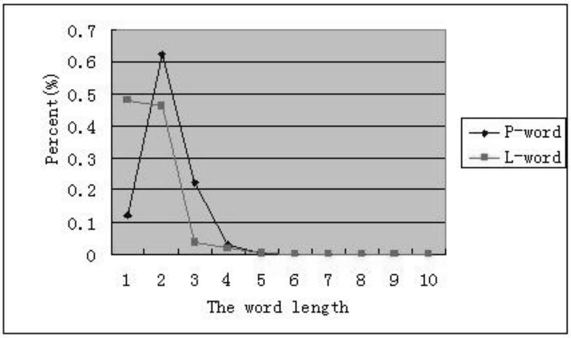


Fig. 1. The length distribution of P-word and L-word

L-words, the P-words can be divided into two classes: the P-words by grouping the L-words and the P-words by splitting the L-word.

2.1 Grouping the L-Words into the P-Word

In the analysis corpus, about 98% P-words are formed by one or more L-words. Figure 2 gives the distribution of the number of L-words to combine together into one P-word. It is shown that most of P-words are from one or two L-words. Figure 3 gives the percentage of each word-length group to form a P-word from two adjacent L-words, where we call the word-length group as the rhythm pattern.

For example, in the sentence, "把 脚趾 插进 沉重 而 潮湿的 沙里", the P-word "沙里" is formed by two mono-syllable L-words of "沙" and "里", which is a "1+1" rhythm pattern. And the P-word "潮湿的" is composed of a bi-syllable L-word and a mono-syllable L-word, which is a "2+1" rhythm pattern. Figure 3 shows that the "1+1" type P-word occupies 43% in the all two L-words grouping. And, when the bi-syllable L-words are adjacent to mono-character L-words, it is also more likely to be assembled. It can be seen that the mono-syllable L-words are the most important candidate to be assembled with others to build the P-words. Almost all P-words formed by grouping include a mono-syllable L-word. We name the constraint of the rhythm pattern demands as the rhythm constraint.

Besides the rhythm constraint, the lexical information is another important cue for grouping the L-words to the L-word, the auxiliary word "的" are more likely attached to the previous word to form a prosodic word. Some other auxiliary words, such as "了", "着", "过", also have this characteristic. Here, the lexical information is mainly represent by the word and part-of-speech.

2.2 Splitting the L-Word into the P-Words

On the other hand, the L-word with more than 3 syllables, are usually divided into several shorter P-words.

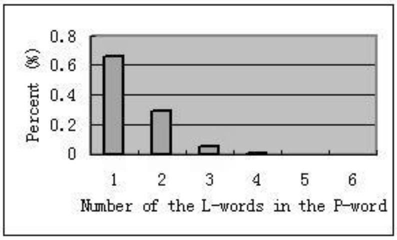


Fig. 2. Distribution of the number of L-words to form a P-word by grouping

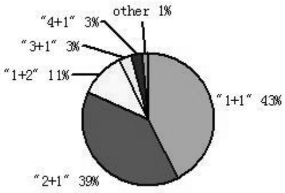


Fig. 3. The percentage of each grouping type of P-words by grouping two L-words

For example, in the sentence "以 行云 流水 般的 演唱 和 精彩的 武工 技艺 赢得了 满堂 喝彩", the lexicon words of "行云流水", "满堂喝彩", are all split into two bi-syllable P-words in the utterance.

The rhythm constraint still plays an important role. From the corpus, the even position in the L-word is the most likely to be split as a P-word boundary. Table 1 gives the occurrence frequency of the splitting position, where w-4 means the L-word with 4 syllables, and w-5 is 5 syllable long, and so on. We can see that in table 1 the separation is more likely to appear at the even position, especially at the second syllable of the word. The position of the character is the main cue for the P-word forming.

Table 1. The splitting position in the long words

	1	2	3	4	5	6
w-4	8	794	5	–	–	–
w-5	2	27	12	2	–	–
w-6	1	3	13	3	0	–
w-7	0	8	7	4	2	0

2.3 Prosodic Word Prediction from the Lexicon Word

From the analysis in this section, we can divide the process of the P-word prediction into two parts, part 1: combining the L-words into a P-word, which is called as the word grouping model; part 2: separating some L-word into the P-words, which is named the word splitting model. In the word grouping model, the model will check every L-word boundary and decide whether it is a P-word boundary. In the word splitting model, the model will only check the L-word longer than 3 syllables, and decide whether the character boundary in this L-word is the prosodic word boundary. These two tasks can be all considered as the binary classification problem. In this paper, we use the maximum entropy model as the solution to model the word grouping and the word splitting process, respectively.

3 Maximum Entropy Based Prosodic Word Prediction

3.1 Maximum Entropy Model and Feature Selection

The maximum entropy(ME) approach had been introduced to many tasks of Natural Language Processing[6], and had achieved a good performance for some problems, like Chinese word segmentation[7], and statistic machine translation[8]. The ME model is good at the classification problem.

$$p(y|x) = \frac{1}{Z_\lambda(x)} \exp(\sum_i \lambda_i f_i(x,y)) \tag{1}$$

$$Z_\lambda(x) = \sum_y \exp(\sum_i \lambda_i f_i(x,y)) \tag{2}$$

Formula 1 and formula 2 gives the computation formula for the probability model described by the ME approach. In these formulas, x is the context that can be observed, and y is the class to be predicted. $p(y|x)$ represents the conditional probability of the class y given the context of x. As the formula 3 shown, the class with the maximum conditional probability given the context or observation will be the prediction result.

$$y^* = \arg\max_y p(y|x) \tag{3}$$

Where, $f_i(x,y)$ is the feature function, and λ_i is the weight parameter for this feature. There are several parameter estimating methods for the λ_i, such as Generalized Iterative Scaling(GIS) [9], Improved Iterative Scaling(IIS) [6], L-BFGS. In this paper, we use the GIS method for the parameter estimation. One of the ME method's advantages is that the model can directly use many different features and doesn't need to assume the independence between them. Any feature related to this problem can be added to the model. This will lead to so many features to the ME model. Sometimes, the feature number can reaches several million. However, of all the features, many features are not useful for the model at all. On the other hand, some features are more important in the model. So, selecting a suitable feature set from millions of features become an important work for the ME model. There are two methods of feature selection for the ME model: cutoff based feature selection and the likelihood gain based feature selection. Cutoff method is to remove the features occurring less times[10]. And the likelihood gain based method is to select the feature with maximum contribution to the likelihood of the ME model every time[11]. These two methods are all used in this paper.

3.2 ME Based Word Grouping Model

The word grouping model is used to decide whether the lexicon word boundary should be the prosodic word boundary. It's a binary classification problem, where '1' means that this L-word boundary is a P-word boundary, and '0' means that it is not a P-word boundary, or, the two L-words beside this point should be combine together in a P-word. Table 2 gives the feature template for the word grouping model. According to the analysis in section 2, three classes of features are used in the feature template. Class 1 is about the word feature[1]; class 2 concerns the POS feature; class 3 is about the word length, which represents the rhythm information. And the other combined features are also included, like the combination of the POS and word length.

In the word grouping model, the cutoff based feature selection is firstly used. We set the cutoff at 10, which will remove the features occurring less than 10 times in the training set. After cutoff, the feature inducing method in [11] is used to select the efficient features for the word grouping model, which select the feature with maximum log-likelihood gain of the training set every time. The developing set is used to avoid the overtraining in the training set. By this feature inducing method, a final feature set is selected. The next section will give the process of feature selection.

Table 2. The feature template for the word grouping model

word(w)	POS tag(t)	word length(l)
w_{-2},w_{-1},w_0,w_1,w_2, (w_{-1},w_0), (w_0,w_1), (w_0,t_1), (t_0,w_1)	t_{-2},t_{-1},t_0,t_1,t_2, (t_{-1},t_0), (t_0,t_1), (t_0,t_1)	l_{-2},l_{-1},l_0,l_1,l_2, (l_{-1},l_0), (l_0,l_1), (l_1,l_2) (t_{-1},l_{-1}), (t_0,l_0),(t_1,l_1), (t_0,l_0,t_1,l_1)

3.3 ME Based Word Splitting Model

The prosodic foot structure in the lexicon word can be added to the lexicon, like in [1], which can provide the prosodic word information in the L-word. But it needs much work of labelling the lexicon. And, the out-of-vocabulary(OOV) will lead to a new problem. The lexicon will not include the OOV's prosodic word information. So, building a word splitting model becomes necessary.

The splitting model is used to check every character in the long word, and decide whether it is split as a P-word boundary at this character boundary, where '0' means that this character boundary is not a P-word boundary and '1' means that it is a P-word boundary or this L-word will be split at this point. This is also a binary classification problem. The splitting model is used, only when the lexicon word is longer than 3. According to the analysis in section 2, we select the feature template shown as table 3. The template includes two kinds of features. One is

[1] In the feature templates, the subscript denotes the position of the word. For example, w_0 denotes the current word, and w_{-n} or w_n denotes the word n position to the left or right of the current word. This subscript is also used to denote the POS tag, the word length and the character's position in the word splitting model. In the splitting model, the subscript denotes the position of the character in the word.

about the character information. It is important, because the characters in a lexicon word can form some little sub-words, which may be a P-word, too. Where, $T(c_0c_1)$ denotes whether the character group of c_0 and c_1 is a lexicon word. When c_0c_1 is a lexicon, $T(c_0c_1)$ equals to '1'. Otherwise, $T(c_0c_1)$ equals to '0'. We judge it by looking up a lexicon. Another kind of feature is about the position information, where the p_0 means the position of the current character, dist_e is the distance from the current character to the end of the word. And, the features combining the character and the position information are also used, like (c_0,p_0).

In the corpus, only when the lexicon word is longer than 3, called as 'the long word', will it be split. There are only 742 long words in the training set, 114 long words in the developing set. and 150 long words in the testing set. 114 words seems not enough data for the developing set. So, we only use the cutoff method to select the features, and did not use the feature inducing. Here, we set the cutoff value as 5. In order to avoid the training data sparse, the training set(4000 sentences) and developing set(1000 sentences) are used together for training in this splitting model. And the parameter estimation is made using the GIS method[9].

Table 3. The feature template for word splitting model

character(c)	position(p)
c_{-1}, c_0, c_1, (c_{-1},c_0), (c_0,c_1), (c_1,c_2) $T(c_0c_1)$, $T(c_1c_2)$	p_0, dist_e, $(p_0,\text{dist\_e})$, (c_0,p_0), (c_{-1},c_0,p_0), (c_0,c_1,p_0), (c_1,c_2,p_0)

4 Experiments and Analysis

Firstly, the feature inducing is made for the word grouping model. The figure 4 gives the change in the log-likelihood of the training set and the developing set during the feature inducing. When inducing the features, the feature with the maximum log-likelihood gain on the training set is selected each time[6]. The blue line with '*' is the result on the training set. And the red line with 'o' is the result on the developing set. With the features being added, the likelihood for the training set keeps increasing all the time. However, the likelihood gain of the developing set begins to stop at about 1250th feature. It's to say the overtraining begins at this time. Therefore, we choose the top 1250 features as the feature set and estimate the parameter for each feature using the GIS algorithm. There are totally 5637 feature candidates. The size of the induced feature set is not up to the 1/4 of all the feature candidates. With this feature set, the testing result on the testing set is given in table 5, where we abbreviate the word grouping model to WG. The precision and recall[2] reaches 93.67% and 97.51%, respectively. And, it is worth mentioning that the precision and recall

[2] The precision and the recall in the prediction of the word grouping and the word splitting model is computed based on the prosodic word boundary.

is 93.36% and 97.02% when using all the candidate features as the feature set. While the feature set is reduced, the model's performance is even improved. It's to say, this feature inducing method is efficient in both reducing the size and improving the performance of the model.

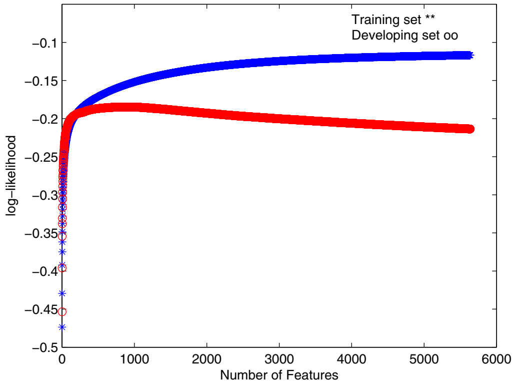


Fig. 4. The change in log-likelihood with the feature selection

Table 4 gives the top 10 features of the induced feature set for the word grouping model. Where, the weight λ with the positive value means that this feature support this class y and the negative feature disprove the this class result. And, the value of λ represents the degree. For example, the weight of the first feature is 1.0077. It means that when the following L-word is a two-syllable L-word, the current L-word boundary is 2.7394 ($= e^{1.0077}$) times more likely to be a P-word boundary than when the following L-word is not a two-syllable L-word. In this feature set, the first two features mean that the bi-syllable word is hard to be combined with its previous word. The third feature is to say the auxiliary word with the one-syllable length is hard to be separated from the previous word. According to our experience, it is right, because the auxiliary words, like '的', '了', '着', are always attached to the previous word to form one prosodic word. The 4th feature is that when the current L-word is two syllable long, it is also hard to be combined with the next L-word. However, the value of the feature is not high, which account for a part of the bi-syllable L-word can be attached with other L-words, such as the word '的'. The 5th feature is to say that the last name and the first name of the Chinese name with three syllables are always combined together as a P-word. All the features are reasonable.

Table 5 also gives the results for the word splitting(WS) part respectively, which only use a cutoff of 5 to select the features. And the f-score of the splitting model reaches 96.67%. We combined the word grouping and word splitting(WG & WS) together to get the final result of the P-word prediction. The f-score of the whole model reaches 95.55%. The baseline is to use the lexicon word directly as the prosodic word. The results by the ME model has reached or outperformed

Table 4. The top 10 features in the word grouping model

No.	Feature f	λ
1	l1=2 → 1	1.0077
2	l1=2 → 0	-1.2633
3	t1=u and l1=1 → 1	-2.8447
4	l0=2 → 0	-0.5781
5	t0=nr and l0=1 and t1=nr and l1=2 → 0	2.6563
6	w0=的→ 0	-3.4568
7	l1=3 → 0	-2.6890
8	t1=b and l1=1 → 1	-2.4421
9	t1=q and l1=1 → 1	-2.0262
10	l1=4 → 0	-3.1836

Table 5. The test results

	precision(%)	recall(%)	f-score(%)
Word grouping(WG)	93.63	97.51	95.53
Word splitting(WS)	96.67	96.67	96.67
WG & WS	93.69	97.49	95.55
Basline	72.88	98.04	83.61

the state of the art reported in [1] and [3]. The ME approach is competent for the P-word predicting task.

5 Conclusions

In this paper, we study the difference between the P-word and the L-word, and propose a P-word prediction approach through grouping and splitting the L-word. Besides the lexical information, the research discovers that the rhythm pattern plays an important role to constrain the prosodic word, which trends to be two syllables long. And the mono-syllable word is easy to be attached to the previous word. The ME approach is introduced to model the word grouping and the word splitting in the P-word prediction. In the ME framework, many different features can be used regardless of the features' dependence. By using the likelihood based feature inducing, the useful features are selected, which decrease the model size while keeping the performance. It is shown that much more efficient features are selected firstly by this feature inducing algorithm. The experiments show that the ME model is competent for the P-word prediction. The f-score of the prosodic word prediction reaches 95.55%.

It must be mentioned that the ME model assumes that the P-words are independent between each other in the sentence, and each P-word is decided by itself. Actually, the interaction between the P-words exists. In the future work, we will focus on how to model the dependence between the P-word boundaries.

References

1. Qian, Y., Chu, M. and Peng, H., 2001. Segmenting unrestricted Chinese text into prosodic words instead of lexicon words, Proceeding of the 2001 International Conference on Acoustic, Speech and Signal Processing, 2001, Salt Lake City.
2. Qin Shi and XiJun Ma, 2002. Statistic Prosody Structure Prediction., Int. Proc. of the IEEE 2002 Workshop on Speech Synthesis, Santa Monica, Ca., 2002.
3. Zhao Sheng, Tao Jianhua, etc, 2003. Chinese prosodic phrasing with extended features, ICASSP2003.
4. Jianfen Cao, The Rhythm of Mandarin Chinese, Journal of Chinese Linguistics, Monograph Series 17, University of California, Berkeley, USA. 2002.
5. Wang Hongjun, 2000. Prosodic words and prosodic phrases in Chinese, Chinese languages and writings, Volume 274-279, 2000
6. Berger, A. L., Della Pietra, V. J., Della Pietra, S. A. A Maximum Entropy Approach to Natural Language Processing. Computational Linguistics, 1996, 22(1): 39-71.
7. Low, Jin Kiat, Ng, Hwee Tou, Guo, Wenyuan, 2005. A Maximum Entropy Approach to Chinese Word Segmentation. Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing. (pp. 161-164). Jeju Island, Korea.
8. Franz Josef Och, Hermann Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. Proc. of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 295-302, Philadelphia, PA, July 2002.
9. Darroch, J. N., Ratcliff, D. Generalized Iterative Scaling for Log-Linear Models. The Annals of Mathematical Statistics, 1972, 43(5): 1470-1480.
10. Ratnaparkhi, A. MAXIMUM ENTROPY MODELS FOR NATURAL LANGUAGE AMBIGUITY RESOLUTION. Computer and Information Science, University of Pennsylvania, 1998.
11. Della Pietra, S., Della Pietra, V., Lafferty, J. Inducing features of random fields. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1997, 19(4): 380-393.

Predicting Prosody from Text

Keh-Jiann Chen[1], Chiu-yu Tseng[2], and Chia-hung Tai[1]

[1] Institute of Information Science, Academia Sinica, Taipei
[2] Phonetics Lab, Institute of Linguistics, Academia Sinica, Taipei
kchen@iis.sinica.edu.tw, cytling@sinica.edu.tw,
glaxy@iis.sinica.edu.tw

Abstract. In order to improve unlimited TTS, a framework to organize the multiple perceived units into discourse is proposed in [1]. To make an unlimited TTS system, we must transform the original text to the text with corresponding boundary breaks. So we describe how we predicate prosody from Text in this paper. We use the corpora with boundary breaks which follow the prosody framework. Then we use the lexical and syntactic information to predict prosody from text. The result shows that the weighted precision in our model is better than some speakers. We have shown our model can predict a reasonable prosody form text.

1 Introduction

In order to improve the prosody of unlimited TTS, a framework to organize the multiple perceived units into discourse is proposed in [1]. Some preceding study regards fluent speech as a succession of independent sentences. If we only apply succession of discreet and often declination intonations to unlimited Mandarin Chinese TTS (text-to-speech synthesis), the unlimited TTS can not produce satisfactory fluent speech prosody. However in our framework, these units are not equal for perception. Some perceived units are grouped by a higher-level unit. The higher-level unit governs and constrains the lower-level units. Lower-level units in different position presented different acoustic patterns rather than being regarded as the same prosodic unit. In other word, this is a hierarchical framework. As Figure 1 illustrated, these units located inside different levels of boundary breaks across speech flow. The boundaries are annotated using a labeling system that annotated small to large boundaries with a set of five break indices. i.e., B1-B5. The framework can also be viewed as a tree-branching organization of multi-phrase prosody.

From bottom up, the layered nodes are syllables (SYL), prosodic words (PW), prosodic phrases (PPh) or utterances, breath group (BG) and prosodic phrase groups (PG). These constituents are, respectively, associated with break indices B1-B5.

B1 denotes syllable boundary at the SYL layer where usually no perceived pauses exist. B2 is a perceived minor break at the PW layer. B3 is a perceived major break at the PPhs layer. B4 denotes a boundary break when the speaker is out of breath and takes a full breath and breaks at the BG layer. B5 is when a perceived trailing-to-a-final-end occurs and the longest break follows. Table 1 shows the definition of all

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 179 – 188, 2006.

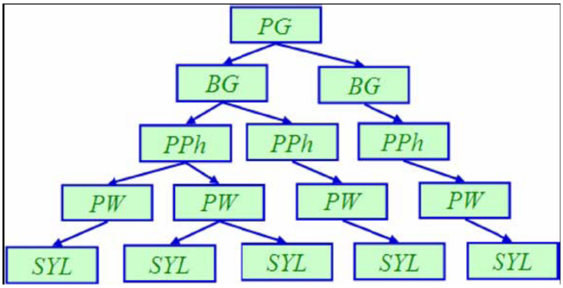
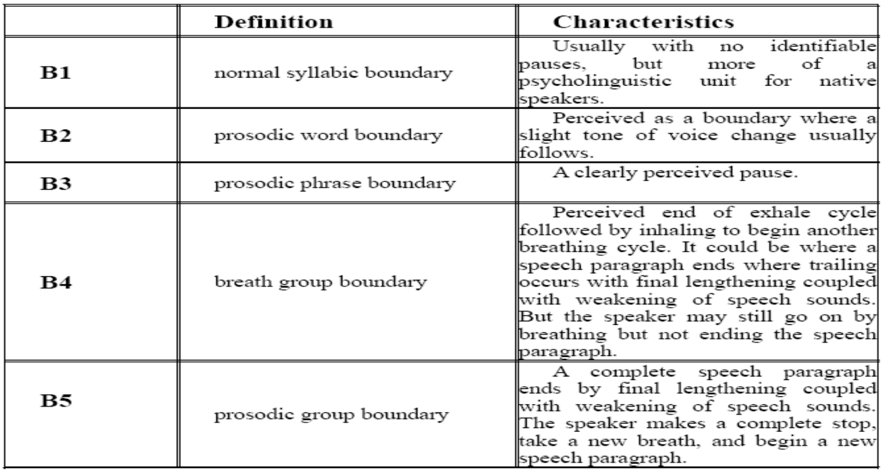


Fig. 1. A schematic representation of the prosody framework

Table 1. Index of Break Hierarchy and Transcription Consistency



	Definition	**Characteristics**
B1	normal syllabic boundary	Usually with no identifiable pauses, but more of a psycholinguistic unit for native speakers.
B2	prosodic word boundary	Perceived as a boundary where a slight tone of voice change usually follows.
B3	prosodic phrase boundary	A clearly perceived pause.
B4	breath group boundary	Perceived end of exhale cycle followed by inhaling to begin another breathing cycle. It could be where a speech paragraph ends where trailing occurs with final lengthening coupled with weakening of speech sounds. But the speaker may still go on by breathing but not ending the speech paragraph.
B5	prosodic group boundary	A complete speech paragraph ends by final lengthening coupled with weakening of speech sounds. The speaker makes a complete stop, take a new breath, and begin a new speech paragraph.

breaks and the characteristics of those. When acoustic parameters of unlimited TTS are strung into speech flow, they must adjust and modify to derive satisfactory fluent speech prosody. How acoustic parameters adjust and modify is according to which level of boundary breaks they located inside. To make an unlimited TTS system, we must transform the original text to the text with corresponding boundary breaks. So we describe how we predicate prosody from Text in this paper.

To predict prosody from text we need the corpora with boundary breaks. We describe the corpora we used in more detail in Section 2. The prosody production models are described in the section 3. The section 4 shows experimental results.

2 Materials Used--Text vs. Speech Corpora

COSPRO 01 and 05 speech data from Sinica COSPRO Database [2] were used. COSPRO 01 contains 599 paragraphs (24803 syllables in total) ranging from 2-character simple sentences up to 181-character complex sentences. COSPRO 05 consisted of readings of 26 paragraphs (11592 syllables in total) of text ranging from 85 to 981 characters per paragraph rearranged from the COSPRO 01 for frequency and phonetic controls. The two sets of text overlapped 88%. Four native untrained speakers (2 males M01, M02 and 2 females F01, F02) read the COSPRO 01 at the average speech rate of 304 ms/syllable in COSPRO 01. Another two radio announcers (1 male and 1 female) read the 26 longer paragraphs at the average speaking rate of 200 ms/syllable in COSPRO 05. Segmental identities were first automatically labeled using the HTK toolkit and SAMPA-T notation, then hand tagged by trained transcribers for perceived boundary breaks using the Sinica COSPRO Toolkit [3]. All labeling was also spot-checked by trained transcribers.

The majority of PWs were disyllabic (67%) and tri-syllabic (25%) [4]. Although the length of PPhs are mostly under 10, the variations of PPhs were more complicated than PWs. Figure 2 shows the distribution of the length of PPhs in COSPRO 01.

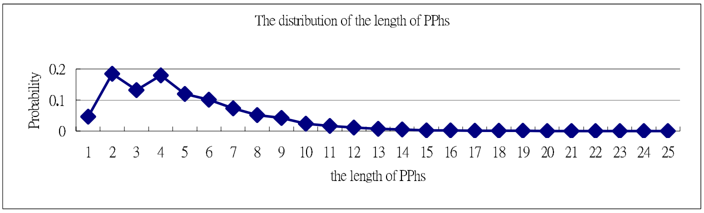


Fig. 2. The distribution of the length of PPhs

The length of PPhs seems not a suitable feature to predict the PPhs. Instead syntactic structures are somewhat related to the structures of PPhs. They do have some common patterns shown in the prosodic structure annotated speech data and syntactic annotated text. For instance, the prosody structure of the sentence “中油公司高級主管昨天表示” is shown in Figure 3 and its syntactic structure is shown in Figure 4. The first PPh is coincident with the NP structure and the second PPh is a partial VP structure. Our predicting models are trained from the prosodic and syntactic structure aligned parallel corpora. We will present our prediction models in more details in Section 3.

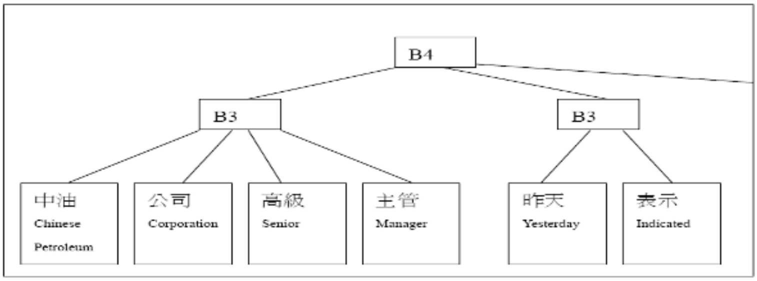


Fig. 3. Part of prosody structure for "中油公司高級主管昨天表示"

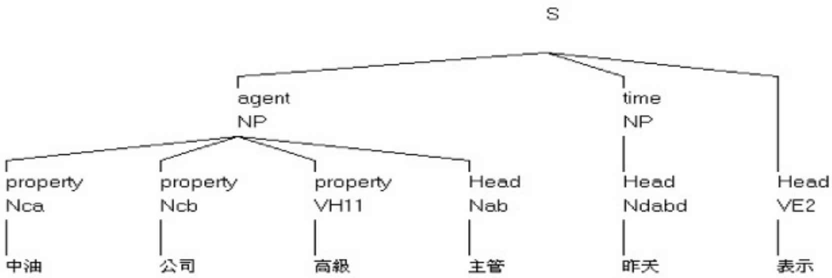


Fig. 4. Syntactic structure of "中油公司高級主管昨天表示"

3 The Model for Predicting Prosody from Text

We propose a series of bottom up models to predict prosody from text. We use word segmentation program (http://rocling.iis.sinica.edu.tw/CKIP/wordsegment.htm), POS tagger and Chinese parser [5] to retrieve the syntactic and lexical information of sentences for training and applying our models. The major features used in our models include lexical words, part-of-speeches (POS), syntactic structures, and lengths. Prediction of B1 is obvious, since character boundaries are natural boundaries of SYL in Chinese. For predicting PWs, length of the word and POS are two essential features. Since there is no gold standard for PW, a consistency checking with human speeches is performed. An average performance of 90% F-score is achieved for PW prediction. Comparing with the average consistency F-score of 92% among human speakers, the model performs quite well. The detail PW model is in [4].

For PPh prediction, A conditional probability $P(B3 | Ph, PL, MPhYN, B, X)$ of a location X to be a PPh boundary B3 was proposed to model the production of PPhs. Where the conditional feature *Ph* is the name of the phrase contained the prosodic word at left of *X*. *PL* is the length of *Ph*. *MPhYN* is a value of yes/no which indicates whether the *Ph* is an embedded phrase or not. *B* is the boundary type of *X*. There are four different types. They are "| |", "| (", ") |", and ") (". "| |" means that the PWs in the both sides of *X* are in the same phrase. "| (" means that *X* is the left boundary of an embedded phrase. Similarly, the ") |" means that *X* is the right boundary of an embedded phrase. The ") (" means that *X* is located between two embedded phrases.

Table 2. The occurrence probabilities of B3 at different types of boundaries

Boundary representation	The probability of PPh
| |	0.214669
| (	0.316559
) |	0.380176
) (	0.589354

The probabilities of being a PPh boundary for different boundary types observed from COSPRO corpus are demonstrated in Table 2. The probability of PPh in “) (” is much higher than others which means that having a PPh break between two complete syntactic units is preferred.

$P(B3 | Ph, PL, MPhYN, B, X)$ can be derived from annotated training corpora by Maximum-likelihood or Maximum Entropy estimations. The complete PPh production model is shown below.

PPh Production Model:

Input: A sequence of sentences with word, POS, PW and syntactic structure annotated.

Algorithm: For each input sentence,

Step 1. Assign B3 to every place with punctuation markers of comma, period, question mark, exclamation mark, and semicolon.

Step 2. For each PW boundary X, derive the value of $P(B3 | Ph, PL, MPhYN, B, X)$.

Step 3. Determine the number of PPhs m in the input sentence by a control parameter n which is an integer value proportional to the intended speech rate.

m=[Length of sentence/n] ,where n are usually set to 5 or 7 for normal speed.

Step 4. Assign m number of B3 at X1,X2,…,Xm which have the highest accumulated probabilities of $\sum_{1}^{m} P(B3 | Ph, PL, MPhYN\ , B, Xi)$, such that no resulting PPh contains only single PW.

Figure 5 shows the algorithm of producing complete prosody. In Step (1) of the algorithm, we read in a text of multiple paragraphs with punctuations. In Step (3) we use a PW model [4] to predict PW boundaries B2. For long sentences, which are longer than 10 characters, the PPh production algorithm will be applied to mark B3. After we decide PPhs, at step (5) we mark B5 before identify breath group BG, since the location of a B4 depends on the length of PG and speech rate. Since PG is a discourse unit and usually is a complete paragraph, naturally we use periods and question marks to predict PG. On the other hand, BG is caused by physical

```
    The Procedure of prosody prediction
(1)   Input Data= text of multiple paragraphs
(2)   Text with word boundaries, POS and syntactic structure annotation
      is produced by a syntactic parser.
(3)   Identify PWs for each Sentence in Input Data
(4)      If (SentenceLength(Sentence)>10) then
            apply PPh model to identify PPhs.
(5)   Identify PG.
(6)   Identify BG.
(7)   Output Data: text of multiple paragraphs with boundary breaks
```

Fig. 5. The algorithm for predicting prosody from text

constrain of human exhale cycles. It is obvious that predicting of breath groups depends on speech rate and length of PGs. Normally, 20~30 syllables are produced in each exhale cycle. Table 3 shows the statistics of the 4 speakers on COSPRO 01 data. Within a long PG, we need to find natural stopping points for inhale and next exhale cycle. For every PG, we use following heuristic rules to mark B4 in the step (6).

(1) Every end of a sentence is a possible candidate of B4 and obviously B5 is mandatory a B4.
(2) For each B4 candidate, if the number of characters to the next B5 is greater than 40 or the followed sentences has more than 30 characters, then we mark it as B4.

After those steps, we had text of multiple paragraphs marked with different levels of boundary breaks as output file. Then the prosody of the text is established by the boundary breaks.

Table 3. Statistics of the lengthes of BGs of the corpora COSPRO 01 and 05

Corpus	Speaker	Maximum	Minimum	Average	Most
COSPRO01	F01	92	3	25.5	23
	F02	104	8	32.3	23
	M01	148	1	27.5	23
	M02	109	3	22.3	17
COSPRO05	F051	133	6	29.8	25

4 Experimental Results and Evaluations

Cross-validation was applied on the data COSPRO 01. The COSPRO 01 was split into six subparts 100 paragraphs each. Each subpart was tested in turn with other 5 subparts as training data. We also used COSPRO 05 as testing data for open test.

4.1 Evaluation Metrics

To evaluate the performances of prediction models, we propose three different sets of evaluation metrics. Each set of evaluation metrics consists of recall, precision, and balanced F-score, the harmonic mean of precision and recall, but with slightly different senses.

$$\text{Precision} = \frac{\text{number of correctly predicted boundary breaks}}{\text{number of predicted boundary breaks}}$$

$$\text{Recall} = \frac{\text{number of correctly predicted boundary breaks}}{\text{number of real boundary breaks}}$$

$$Balanced\ F-score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

The first set of evaluation metrics takes each human performance as standard. As a result it shows the degree of consistency between machine and human performances. We proposed a second evaluation metric called weighted precision to evaluate the quality of our prediction. The idea is that more speakers agree upon the boundary break which gets more weight. If one speaker agrees with the position, we give the weight 0.25. If two speakers agree with the position, we give the weight 0.5. If all four speakers agree with the position, we give the full weight 1. The third set of metrics of evaluation is called "general precision" which a prediction of break type matching any one of speaker is considered correct.

4.2 Evaluation Results and Analysis

We evaluate the performances of each individual model and compare them with human produced prosody. The first section is the evaluation results of PPh model, and the next section contains evaluations for B4 (breath groups) and B5 (prosodic phrase groups).

4.2.1 The Evaluation Results of PPh Model

PPh model was applied by controlling parameter of speech rates at two different values *n*=5 and 7. The results of cross validation on COSPRO 01 with respect to four different speakers F01, F02, M01, and M02, are showed in Table 4 and Table 5. The F-Score of our model is around 73%. We also calculate the consistency among speakers. The consistency among four speakers of prosodic phrases on COSPRO 01 is shown in Table 6. The F-Score of speaker's consistency on COSPRO 01 are around 75%. The Results show that our model performs almost comparable with the human speaker's consistency.

Table 4. The result of n=5 for PPhs on COSPRO 01

	F01	F02	M01	M02
Recall	0.6990	0.6966	0.7913	0.6628
Precision	0.7782	0.7815	0.6632	0.7918
F-score	0.7365	0.7366	0.7216	0.7216
Weighted Precision General Precision	0.80 0.88			

Table 5. The result of n=7 for PPhs on COSPRO 01

	F01	F02	M01	M02
Recall	0.6679	0.6587	0.7715	0.6258
Precision	0.8423	0.8413	0.7342	0.8486
F-score	0.7450	0.7389	0.7524	0.7204
Weighted Precision General Precision	0.85 0.92			

Table 4 and 5 also show the weighted precision in different *n*, and the weighted precisions are over 80%. In Table 7, the lowest weighted precision of four speakers in CORPOS 01 is 80%, and the weighted precision in our model is comparable with human speakers. Regarding the general precision of our model, over 88% of our predictions are marked as PPh by at least one of those four speakers. These evaluation results show that our model performs well and can consistently identify prosodic phrases.

Table 6. The consistency of PPhs among human speakers on COSPRO 01

	F01	F02	M01	M02
Recall	0.789	0.793	0.643	0.815
Precision	0.747	0.726	0.857	0.711
F-Score	0.763	0.754	0.735	0.756

Table 7. The weighted precisions of B3 among human speakers on COSPRO 01

F01	F02	M01	M02
0.83811	0.82513	0.91738	0.80268

We use the COSPRO 05 for our open test. Table 8 shows the evaluation results of PPh model in comparing with two speakers M051 and F051 at different speech rates *n*. The F-Score of our model is around 78%. It is close to the F-Score of human speaker's consistency of 80% shown in the Table 9. Because there are only two speakers in COSPRO 05, we do not evaluate the weighted precision and general precision.

Table 8. The evaluation results of PPhs model at different speech rates on COSPRO 05

M051	Recall	Precision	F-score
n=5	0.7791	0.6398	0.7026
n=10	0.7431	0.7558	0.7494
n=15	0.6972	0.8444	0.7638
F051	Recall	Precision	F-score
n=5	0.7945	0.6444	0.7116
n=10	0.7644	0.7678	0.7661
n=15	0.7280	0.8707	0.7930

Table 9. The human speaker's consistency on PPhs production at COSPRO 05

	Recall	Precision	F-Score
M051-Based	0.801	0.811	0.806

4.2.2 The Evaluation Results of Predicting BGs and PGs

Because COSPRO 01 is not composed by complete text units, we use only COSPRO 05 to evaluate. Table 10 shows the results of BGs on COSPRO 05, and the F-scores are around 55%-60%. The human speaker's consistency of BGs in COSPRO 05 shown in Table 12 is about 0.59. The inconsistency of BGs may be due to the physical difference between the human speakers and the broader scope of BGs. The variation of BGs makes the difficulty of prediction. Our predictions of BGs are close to the human speaker's consistency of BGs.

Table 10. The results of BGs prediction on COSPRO 05

	Recall	Precision	F-score
B4-M051	0.5723	0.5360	0.5535
B4-F051	0.6064	0.5994	0.6028

Table 11 shows the result of PGs on COSPRO 05. The human speaker's consistency of PGs on COSPRO 05 is 63%. Compare to Table 12, the F-score of our prediction is much lower than the F-Score of consistency. The main reason is we do not have paragraph mark in the text. So we mark every period punctuation as prosodic phrase group. It results in the low precision in PG prediction. Another reason may be that the trained transcribers used not only text information but also acoustic information. We only use text information, so the precisions of our prediction are much lower.

Table 11. The result of PG predictions on COSPRO 05

	Recall	Precision	F-score
B5-M051	0.7822	0.3222	0.4564
B5-F051	0.75	0.3388	0.4668

Table 12. The human speaker's consistencies of BGs and PGs at COSPRO 05

	Recall	Precision	F-Score
B4	0.609	0.577	0.592
B5	0.669	0.610	0.638

5 Conclusions and Future Works

This is the first attempt to build a model to predict prosody from text. We used the syntactic structure of text to predict prosodic phrase and used heuristic rules and punctuations to predict breath group and prosodic phrase group. Because the low consistency means the variety of possibility, it makes the difficulty to predict the boundary breaks. Our weighted precision for PPhs on COSPRO 01 is better than some speakers. We have shown our model can predict a reasonable prosody form text. Although we have predicted the prosody model from text, how to use semantic information to group prosodic phrase group is another way to improve our predictions. Using semantic information to predict end of paragraph may help to predict prosodic phrase group. Because we only use punctuation information to determine the end of paragraph, how to use semantic information to detect the change of topic will be our future research.

References

1. Chiu-yu Tseng, Shao-huang Pin, Yeh-lin Lee, Hsin-min Wang, and Yong-cheng Chen (2005). “Fluent speech prosody: framework and modeling,” Speech Communication, Vol.46, issues 3-4,(July 2005), Special Issue on Quantitative Prosody Modelling for Natural Speech Description and Generation, 284-309.
2. Chiu-yu Tseng, Yun-Ching Cheng and Chun-Hsiang Chang (2005). “Sinica COSPRO and Toolkit—Corpora and Platform of Mandarin Chinese Fluent Speech” Proceedings of Oriental COCOSDA 2005,(Dec. 6-8, 2005), Jakarata, Indonesia, 23-28
3. Sinica COSPRO and Toolkit: http://www.myet.com/COSPRO/
4. HuaJui Peng, Chiching Chen, Chiuyu Tseng, Kehjiann Chen ,”PREDICTING PROSODIC WORDS FROM LEXICAL WORDS--A FIRST STEP TOWARDS PREDICTING PROSODY FROM TEXT”, International Symposium on Chinese Spoken Language Processing, ISCSLP, 2004.
5. Yu-Ming Hsieh, Duen-Chi Yang, and Keh-Jiann Chen. 2005. Linguistically-motivated grammar extraction, generalization and adaptation. In Proceedings of the Second International Join Conference on Natural Language Processing, pages 177-187, Jeju Island, Republic of Korea.

Nonlinear Emotional Prosody Generation and Annotation*

Jianhua Tao, Jian Yu, and Yongguo Kang

National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, P.O.X. 2728, Beijing, 100080
{jhtao, jyu, ygkang}@nlpr.ia.ac.cn

Abstract. Emotion is an important element in expressive speech synthesis. The paper makes the brief analysis on prosody parameters, stresses, rhythms and paralinguistic information in different emotional speech, and labels the speech with rich annotation information in multi-layers. Then, a CART model is used to do the emotional prosody generation. Unlike the traditional linear modification method, which makes direct modification of F0 contours and syllabic durations from acoustic distributions of emotional speech, such as, F0 topline, F0 baseline, durations and intensities, the CART models try to map the subtle prosody distributions between neutral and emotional speech within various context information. Experiments show that, with the CART model, the traditional context information is able to generate a good emotional prosody outputs, however the results could be improved if more rich information, such as stresses, breaks and jitter information, are integrated into the context information.

1 Introduction

Recently, more and more efforts have been made in the research of expressive speech synthesis, among which emotion is a very important element [1, 2]. Some prosody features, such as pitch variables (F0 level, range, contour, and jitter), and speaking rate have already been analyzed [3,4]. There are also some implementations in emotional speech synthesis. For instance, Mozziconacci [5] added emotion control parameters on the basis of tune methods, resulting in higher performance. Cahn [6], by means of a visual acoustic parameters editor, achieved the output of emotional speech with manual inferences. Recently, some efforts have been made using a large corpus. A typical system was produced by Campbell [7], who created an expressive speech synthesis from a five years' large corpus that gave impressive synthesis results. Schroeder[8], Eide[9] generated an expressive TTS engine which can be directed, via an extended SSML, to use a variety of expressive styles from about ten hours of "neutral" sentences. Furthermore, rules translating certain expressive elements to ToBI markup have been manually derived. Chuang[10] and Tao[11] used

* The paper was supported by the National Natural Science Foundation of China under Grant (60575032).

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 189 – 199, 2006.

emotional keywords and emotion trigger words to generate an emotional TTS system. The final emotion state is determined based on the emotion outputs from text-content module. The results were used in the dialogue systems to improve the naturalness/ expressiveeness of the answering speech.

As we see, most of current emotional speech synthesis systems are still based on the linear modification method (LMM) on prosody parameters (some of them are also able to make the voice quality control), except unit selection methods. The LMM makes direct modification of F0 contours (F0 top, F0 bottom and F0 mean), syllabic durations and intensities from the acoustic distribution analysis results. The previous analysis shows that the expression of emotion does not just influence these general prosody features, but also affects more subtle prosodic features, such as stresses, breaks, jitter, etc. With this idea, we annotate the emotional speech in more detailed way, and a CART model which can link linguistic features to the prosody conversion is used. To decrease the dimensionality of output prosody parameters, we also use the pitch target model [12] for output prosody parameters. The model is based on the assumption that "observed F0 contours are not linguistic units per se. Rather, they are the surface realizations of linguistically functional units such as tones or pitch accents."[12] To be able to handle the input context information, we also separate them into two parts, one part is traditionally used for normal speech synthesis, the other part is a kind of emotional prosody related information, which normally can only be marked manually. Experiments show that, with the CART model, the traditional context information is able to generate good prosody outputs for some emotion states, while results could be improved if more rich information, such as stresses, breaks and jitter information, are integrated into the context information. Listening tests also show that there are still some distance between output emotional speech and original one, due to the lack of voice quality control which will be solved in our further research.

The paper is composed of five major parts. Section 2 introduces the corpus with emotion labeling. The acoustic features characteristic of emotions were also analyzed. In this section the paper also describes the traditional linear modification model which uses prosody patterns from the acoustic mapping results directly. Further analysis on emotion and prosody reveals that emotions are closely related to subtle prosody distributions, such as stress, rhythm and paralinguistic information. Section 3 describes the CART model which is used to convert the prosody features from "neutral" to emotional speech. The pitch target model is used as the output parameter in this section. In section 4, the paper provides more discussion on the method introduced in the paper via experiments. Section 5 provides a conclusion of the work.

2 Corpus, Analysis and Annotation

We use 2000 sentences of spontaneous dialog speech of one speaker, which were collected via a call center system in daily life, for our work. Each emotion state ("fear", "sadness", "anger" or "happiness") contains 500 sentences. The both linguistic information and paralinguistic information are well reserved in the speech. Each sentence in our database contains at least 2 phrases. There were 1,201 phrases, and 7,656 syllables in total, so on average each utterance contained 2 phrases.

After the collection, all of collected sentences were also read by a professional speaker with a "neutral" way. The recorded speech is used as the reference of the research between emotional states and "neutral" state.

Utterances were then segmentally and prosodically annotated with pitch marks and phoneme (initial and final) boundaries. The emotional speech differs from the "neutral" speech in various aspects, including intonation, speaking rate and intensities, etc. The distribution of prosodic parameters in different emotions of the corpus are shown in table 1.

Table 1. The distribution of prosodic parameters in different emotions

	Neutral	Fear	Sadness	Anger	Happiness
$F0_{mean}$ (Hz)	135	119	108	152	168
$F0_{bottom}$ (Hz)	86	81	83	95	109
$F0_{top}$ (Hz)	181	165	141	256	238
$D_{syllable}$ (ms)	169	173	198	162	178
E (DB)	65	61	61	76	72

Here, the values indicate the means of F0 mean ($F0_{mean}$), F0 topline ($F0_{top}$), F0 baseline ($F0_{bottom}$), syllabic duration ($D_{syllable}$) and intensity (E). The table partly confirms the previous research[1] that "happiness" and "anger" yield a high F0, while "sadness" generate lower F0 than "neutral", and "fear" is quite close to "sadness". The overlap of F0 mean and F0 topline in different emotions is less than that of F0 baseline. It seems that the F0 mean and topline provide better "resolving power" for perception than the F0 baseline.

In general, the choice of contour would then be more related to the type of sentence, while the pitch level and excursion size of the pitch movements would be more related to the speaker's emotional state. Among all traditional emotional prosody generation methods, linear modification method(LMM) seems to be the most intuitive. The model can be described as follows,

$$y_{n,i} = \alpha_{n,i} \cdot x \qquad (1)$$

x indicates the input prosodic parameters, F0 topline, F0 baseline, F0 mean, syllabic duration and intensity. y denotes their outputs among different emotions. α is the transform scale of the parallel prosodic parameters between "neutral" and emotions as calculated from the training set of the corpus. 'n' denotes the emotional state, i.e. "fear", "sadness", "anger" and "happiness", i indexes the emotion level, i.e. "strong", "normal" and "weak".

2.1 Emotion and Jitter

Someone also point out that F0 jitter was an important parameters for emotional speech [1]. For F0 jitter, normally, a quadratic curve was fitted to a running window of 5 successive F0 values on the F0 contour and then subtracted from that section of the F0 contour. It was calculated as the mean period to period variation in the residual F0 values. Table 1 shows the results from emotions in our corpus.

Table 2. The average results of F0 jitter of emotions

Emotions	Fear	Sad	Angry	Happy
F0 jitter (HZ)	6.2	5.9	8.5	12.6

With the results, we can see that “happiness” has the highest F0 jitter while “neutral” contains the minimum value. During speech synthesis, F0 jitter is realized by a random variation in the length of the pitch periods with an amplitude in accordance to the parameters value. This random variation is controlled by a white noise signal filtered by a one pole low pass filter.

2.2 Emotion and Stresses

There also exists a strong relationship between emotions and stresses [13]. Stress refers to the most prominent element perceived in an utterance rather than literal “semantic focus” which is used to express speaker’s attitudes.

In principle, the changing of stresses from “neutral” speech to “emotional” speech could be summarized into five types, decreasing, weakening/disappearing, boosting, increasing and shifting. Decreasing means the amount of stresses is decreased from “neutral” speech to “emotional” speech. Weakening/Disappearing means all of stresses are weakened, some of them are even lost. Boosting denotes the intensity of stresses is amplified. Increasing means the amount of stresses is decreased. Shifting represents the stress locations are changed among emotions.

Some emotions might have more than one stress changing feature. For instance, in “sad” speech, stresses might disappear while “happy” voice may both increase the number of stresses and magnify them. “anger” may both does the stress shifting and also amplifies the stress located on emotional functional words.

2.3 Emotion and Breaks

Compared with the expression of stresses in different emotions, changing features of prosodic rhythms are not very clear to some extent, but there are still some points need to be noted. In neural speech, the most obvious phenomenon about prosodic tempo is that the pitch value becomes higher in the beginning of prosodic phrase and become lower in the ending of phrase. But in emotional speech, being influenced by emotional functional words, which are defined as the focus of emotion, the rule is sometimes broke up. The pitch values of key words always become very high or very low according to different emotions. Due to the impact of speaking rate in different emotions, the amount of prosodic breaks may decrease with fast speaking rate, such as “angry” and “happy”, while they may increase with slower speaking rate, such as “sadness”.

2.4 Emotion and Paralinguistic Information

Although the prosodic function of conveying the expression of emotion seems to involve both a linguistic and a paralinguistic component [14], paralinguistic information normally does more influence on emotion expression. Distinguishing the contour type from its detailed implementation in terms of pitch level and pitch range may well lead to a distinction between linguistic and paralinguistic value of the intonation variations. This expectation is related to the general assumption of the linguistic value of contour type, and the paralinguistic function of its concrete phonetic realization, such as “grunts”, “breathing, etc. Though the prosody is influenced by paralinguistic information, actually, relations between them are very complicated and far from being discovered. Thus, in our current work, we didn’t use them for the prosody model, however we labeled them in the corpus for the further research.

2.5 Multilayer Annotation

We try to label the phenomena which is related to linguistic features, utterance expression and emotions, non-linguistic features, etc. as much as possible, however not all of them could be directly integrated into the system in the meantime. The labelled information is seperated into the different layers.

Transcription Layer
The collected speech was transcribed orthographically with normalized text expression.

Pronunciation Layer
It records the pinyin information of the speech. Initials and finals are also listed.

Pitch Layer
The layer annotates the detailed pitch marks of the voice.

Segmentation Layer
The layer marks syllable or silence/pause boundaries of each utterance. Initial and final boundaries are also labelled.

Break Layer
In our work, we have four types of prosodic boundaries. They are,

- Break0: syllabic boundary.
- Break1: prosodic word boundary, a group of syllables that are uttered closely.
- Break2: prosodic phrase boundary, a group of prosodic words that has a perceptive rhythm break at the end.
- Break3: sentence boundary, the utterance for a whole speech.

Stress Layer
Here, there are three types of stresses, intonation stress, phrasal stress and (prosodic) word stress.

Paralinguistic Information Layer
The paralingual and non-lingual phenomenon included in labels are as follows: beep, breathing, crying, coughing, deglutition, hawk, interjection, laughing, lengthening, murmur, noise, overlap, smack, sniffle, sneezes, yawn, etc.

To be able to use the corpus for further research, we also used the layers in a number of previous schemes. (Core and Allen, 1997; Di Eugenio et al., 1998; Traum, 1996; Walker et al., 1996; MacWhinney, 1996; Jekat et al., 1995; Anderson et al., 1991; Condon and Cech, 1996; van Vark et al., 1996; Walker and Passonneau, 2001).

Layers from different schemes are grouped according to the similar phenomena that they label. They are,

Speech acts
All of the schemes that we examined annotated the utterances for their illocutionary force. Since this is the layer that contains most information regarding the semantic content of an utterance, this is likely to be where we shall find the most interesting correlations.

Communications status
Communications status indicates whether an utterance was successfully completed. It is used to tag utterances that are abandoned or unintelligible rather than whether the intention of a speech act was achieved.

Topic
Several annotation schemes contain this layer that labels the topic discussed in an utterance. This is usually in task domains where there is a finite number of subjects that will be discussed.

Phases
Some schemes distinguish between dialogue phases such as *opening, negotiation* and *query*. Emotion in dialogue also goes through phases and it is possible that there are boundaries between the phases of emotion that correspond to those tagged using the phase layer.

Surface form
Surface form tagging is used in David Traum’s adaptation of the TRAINS annotation scheme (Traum, 1996) and the Coconut scheme to tag utterances for certain special features such as cue words or negation. It has been shown that certain syntactic features of an utterance may be indicators of emotion.

3 CART Model Based Prosody Generation

To be able to handle the context information, we propose a Classification and Regression Trees (CART) which have been successfully used in prosody prediction. The model could do the prosody mapping from “neutral” speech to “emotional” speech with various context informations. The framework of the model is shown in Fig. 1.

In the model, the input context information is classified into two parts, the context part I is normally used for traditional speech synthesis. It contains,

- Tone Identity (including current, previous and following tones, with 5 categories).
- Initial Identity (including current and following syllables' initial types, with 8 categories).
- Final Identity (including current and previous syllables' final types, with 4 categories).
- Position in sentence (including Syllable position in word, word position in phrase and phrase location in sentence)

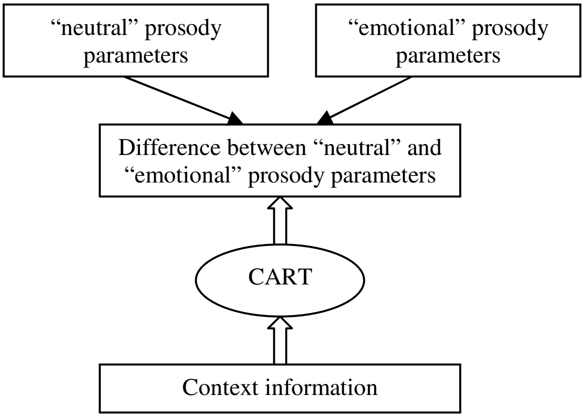

"neutral" prosody parameters
"emotional" prosody parameters
Difference between "neutral" and "emotional" prosody parameters
CART
Context information

Fig. 1. The framework of CART based emotional prosody conversion

- Number (including syllable number of the prosodic word, word number of the phrase, and phrase number of the sentence)
- Part of speech (including current, previous and following words, with 30 categories)

The context part II contains:

- Break types (including intonation phrase boundaries, prosodic phrase boundaries and prosodic word boundaries).
- Stress type (including intentional stress and phrasal stress).
- F0 jitter degree (denote how serious of the F0 jitter in emotional speech).

Since there are lots of changes between "neutral" speech and "emotional" speech in the part II, the information is normally not predicted by text analysis module, but labeled in the input text by markup languages.

The output parameters of the model are the differences of "neutral" and "emotional" prosodic parameters. As we know, Mandarin is a typical tonal language, in which a syllable with different tone types can represent different morphemes. Several models have been proposed to describe F0 contours before, such as the Fujisaki model [15], The Soft Template Mark-Up Language (Stem-ML) model[16], the pitch target model[12] and Title model, etc. In the pitch target model, variations in surface F0 contours result not only from the underlying pitch units (syllables for Mandarin), but also from the articulatory constraints. Pitch targets are defined as the smallest operable units associated with linguistically functional pitch units, and these targets may be static (e.g. a register specification, [high] or [low]) or dynamic (e.g. a movement specification, [rise] or [fall]). With these features, we believe the pitch target model are quite suitable for prosody conversion.

The output parameters are, then, the differences of pitch target parameters a, b, β and λ between "neutral" and "emotional" parameters.

Let the syllable boundary be $[0, D]$. The pitch target model uses the following equations [17].

$$T(t) = at + b \tag{2}$$

$$\begin{gathered} y(t) = \beta \exp(-\lambda t) + at + b \\ 0 \le t \le D, \lambda \ge 0 \end{gathered} \tag{3}$$

Where $T(t)$ is the underlying pitch target, and $y(t)$ is the surface F0 contour. The parameters a and b are the slope and intercept of the underlying pitch target respectively. These two parameters describe an intended intonational goal of the speaker, which can be very different from the surface F0 contour. The coefficient β is a parameter measuring the distance between the F0 contour and the underlying pitch target at t=0. λ describes how fast the underlying pitch target is approached. The greater the value of λ is, the faster the speed. A pitch target model of one syllable can be represented by a set of parameters (a,b,β,λ).

As described in [17], (a,b,β,λ) can be estimated by nonlinear regression process with expected-value parameters at initial and middle points of each syllable's F0 contour. The Levenberg-Marquardt algorithm [17] is used for estimation as a nonlinear regression process.

Wagon toolkit [19], with full CART function, was used in our work. Source and target pitch contours from parallel corpus are aligned according to labelled syllable boundaries, and then pitch target parameters are extracted from each syllable's pitch contour, finally mapping functions of parameters a, b, β and λ are estimated using the CART regression. There were totally four CART models trained with different "neutral" and emotion mappings. For conversion, the pitch target parameters estimated from source pitch contours are transformed by the mapping functions obtained in the training procedure and then the converted pitch target parameters generate new pitch contours associated with the target characteristics.

4 Experiments and Discussion

We used the STRAIGHT[18] model as an acoustic model to generate the emotional speech output with the above CART based prosody model. Here, we didn't do the specific voice quality control in the acoustic level. A prosody converting example is given in Fig. 2.

Eight listeners were asked to give a subjective evaluation on these test sentences. Two methods are conducted to evaluate the proposed emotional conversion:

- ABX test: ABX test in evaluating voice conversion is used in the evaluation. all listeners are required to judge whether a converted speech X sounded closer to a source neutral speech A or a target emotional speech B. This test confirms whether the conversion system is successful.
- EVA test*:* Only converted speeches are listened to and then the associated emotional state is given by these listeners. This test confirms whether the emotional conversion is successful.

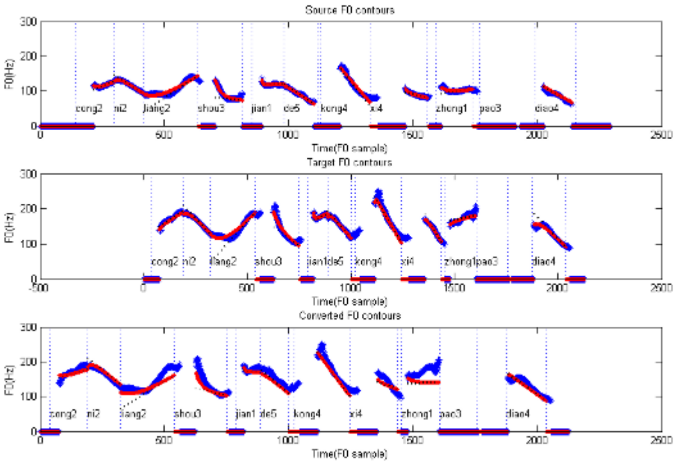


Fig. 2. An example of F0 conversion using the pitch target model in “neutral” to “happiness” conversion with both the context part I and the context part II

Results of the evaluation are shown in Fig.3 and Fig.4, in which X axis is the emotional state and Y axis is the mean correct rate (it is the ratio of judging X as B in ABX test, and considering the converted speech as the corresponding emotional speech in EVA test) of all listeners. ABX test has proved that the converted emotional speech possesses the corresponding emotional state compared with the source speech. Because, in ABX tests, conners can compare the converted speech with the source “neutral” speech, results of ABX tests are better than those of EVA tests. There are differences among emotional conversions in these perception tests, in which the “neutral-sadness” and “neutral-fear” conversions are respectively best and worst. With only context part I, both “fear” and “happiness” are very hard to be simulated, while “sadness” are a little bit easier, since “sadness” is normally related to general

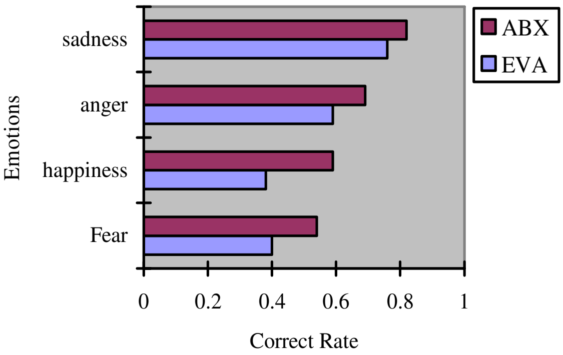


Fig. 3. Results of expressive converting evaluations with the context part I

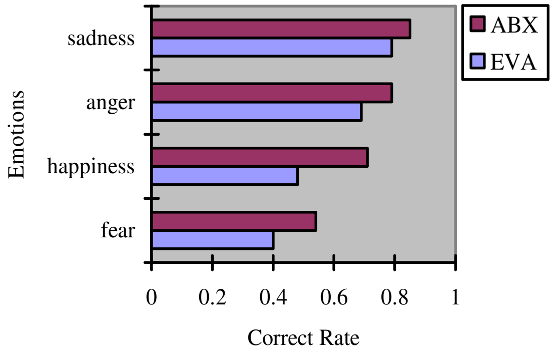


Fig. 4. Results of expressive converting evaluations with both the context part I and context part II

prosodic features, such as narrow pitch range, low pitch level, and slow speed. From Fig.4, it shows that the emotional prosody output will be better if we use all of the context information than that with only the context part I, while more detailed control of prosody information is integrated.

From all of results, we can find none of them get full score in the listening test. Part of reasons might be the lack of voice quality control. The size of the corpus might be another problem for that. Further work will be based on our new collected large corpus (with 2000 sentences for each emotion). More detailed acoustic analysis and voice quality control will also be considered.

5 Conclusion

When generating expressive speech synthesis, we are easily tempted to fall into the practice of using the acoustic patterns driven by the speech with emotion state with a linear modification approach. However, without a more detailed distribution of these acoustic patterns, it is hard for us to synthesize more expressive or less expressive speech. To solve this problem, the paper proposed using a CART method. Unlike the linear modification method, the CART model efficiently maps the subtle prosody distributions between neutral and emotional speech, and allows us to integrate linguistic features into the mapping. A pitch target model which was designed to describe Mandarin F0 contours was also introduced. The experiment results prove that the CART method gives us the very good emotional speech output. The results also show that, with the CART model, the traditional context information which is used for normal speech sythesis, is able to generate good prosody outputs for some emotion states, such as "sadness", while results could be much improved if more rich information, such as stresses, breaks and jitter information, are integrated into the context information. The methods discussed in the paper provide ways to generate emotional speech in speech synthesis, however there is still lots of work to be done in future.

References

1. Murray, I. and Arnott, J. L., “Towards the Simulation of Emotion in Synthetic Speech: A Review of the Literature on Human Vocal Emotion,” in Journal of the Acoustic Society of America, 1993, pp.1097-1108.
2. Stibbard, R. M., “Vocal Expression of Emotions in Non-laboratory Speech: An Investigation of the Reading/Leeds Emotion in Speech Project Annotation Data”. PhD Thesis. University of Reading, UK. 2001.
3. McGilloway, S.; Cowie, R.; Doulas-Cowie, E.; Gielen, S.; Westerdijk, M.; Stroeve S.: Approaching Automatic Recognition of Emotion from Voice: A Rough Benchmark. 2000.
4. Amir, N., “Classifying emotions in speech: a comparison of methods”. Holon Academic Institute of technology, EUROSPEECH 2001, Escandinavia.
5. Sylvie J.L. Mozziconacci and Dik J. Hermes, “Expression of emotion and attitude through temporal speech variations”, ICSLP2000, Beijing, 2000.
6. J.E. Cahn, “The generation of affect in synthesized speech”, Journal of the American Voice I/O Society, vol. 8, July 1990.
7. Nick Campbell, “Synthesis Units for Conversational Speech - Using Phrasal Segments”, http://feast.atr.jp/nick/refs.html
8. M. Schröder & S. Breuer. XML “Representation Languages as a Way of Interconnecting TTS Modules”. Proc. ICSLP Jeju, Korea, 2004
9. E. Eide, A. Aaron, R. Bakis, W. Hamza, M. Picheny, and J. Pitrelli, “A corpus-based approach to <ahem/> expressive speech synthesis”, IEEE speech synthesis workshop, 2002, Santa Monica
10. Ze-Jing Chuang and Chung-Hsien Wu “Emotion Recognition from Textual Input using an Emotional Semantic Network,” In Proceedings of International Conference on Spoken Language Processing, ICSLP 2002, Denver, 2002.
11. Jianhua Tao, “Emotion control of Chinese speech synthesis in natural environment,” in EUROSPEECH- 2003, pp. 2349–2352.
12. Yi Xu and Q. Emily wang, “Pitch targets and their realization: Evidence from mandarin chinese,” Speech Communication, vol. 33, pp. 319–337, 2001.
13. Aijun Li and Haibo Wang, “Friendly Speech Analysis and Perception in Standard Chinese”, ICSLP2004, Kerea, 2004.
14. Laver,J., “The phonetic description of paralinguistic phenomena”, the XIIIth International Congress on Phonetic Sciences. Stockholm, Sweden, Supplement, 1-4, 1995
15. H. Fujisaki and K. Hirose. Analysis of voice fundamental frequency contours for declarative sentence of Japanese. J. Acoust. Soc. Jpn. (E) 5(4):233–242, 1984.
16. Kochanski, G. P. and Shih, C. “Stem-ML: Language independent prosody description”, the 6th International Conference on Spoken Language Processing, Beijing, China.
17. Xuejing Sun, The Determination, Analysis, and Synthesis of Fundamental Frequency, Ph.D. thesis, Northwest University, 2002.
18. Hideki Kawahra, Reiko Akahane-Yamada, “Perceptual Effects of Spectral Envelope and F0 Manipulations Using STRAIGHT Method”, J. Acoust. Soc. Am., Vol.103, No.5, Pt.2, 1aSC27, p.2776 (1998.5)
19. http://festvox.org/docs/speech_tools-1.2.0/x3475.htm
20. Nick Campbell, “Getting to the Heart of the Matter; Speech is more than just the Expression of Text or Language”, LREC, 2001

A Unified Framework for Text Analysis in Chinese TTS

Guohong Fu[1,2], Min Zhang[3], GuoDong Zhou[3,4], and Kang-Kuong Luke[2]

[1] Dept of Chinese, Translation and Linguistics, City University of Hong Kong, Hong Kong
[2] Departmant of Linguistics, The University of Hong Kong, Hong Kong
[3] Institute for Infocomm Research, Singapore 119613
[4] School of Computer Science and Technology, Suzhou University, Suzhou 215006
ghfu@hotmail.com, mzhang@i2r.a-star.edu.sg, gdzhou@suda.edu.cn,
kkluke@hkusua.hku.hk

Abstract. This paper presents a robust text analysis system for Chinese text-to-speech synthesis. In this study, a lexicon word or a continuum of non-hanzi characters with the same category (e.g. a digit string) are defined as a morpheme, which is the basic unit forming a Chinese word. Based on this definition, the three key issues concerning the interpretation of real Chinese text, namely lexical disambiguation, unknown word resolution and non-standard word (NSW) normalization can be unified in a single framework and reformulated as a two-pass tagging task on a sequence of morphemes. Our system consists of four main components: (1) a pre-segmenter for sentence segmentation and morpheme segmentation; and (2) a lexicalized HMM-based chunker for identifying unknown words and guessing their part-of-speech categories; and (3) a HMM-based tagger for converting orthographic morphemes to their Chinese phonetic representation (viz. pinyin), given their word-formation patterns and part-of-speech information; (4) a post-processing for interpreting phonetic tags and fine-tuning pronunciation order for some special NSWs if necessary. The evaluation on a pinyin-notated corpus built from the Peking University corpus shows that our system can achieve correct interpretation for most words.

Keywords: Chinese TTS, text analysis, lexical analysis, grapheme-to-phoneme conversion, text normalization.

1 Introduction

Text analysis for text-to-speech (TTS) synthesis aims to convert a text of orthographic characters into a linguistic representation for speech synthesis. In Chinese, text analysis consists of five sub-tasks, namely word segmentation, part-of-speech tagging, text normalization, word pronunciation (viz. grapheme-to-phoneme conversion, G2P), and prosodic phrasing [1][2]. This study focuses on the first four tasks.

Robust text analysis is essential to develop a high-quality TTS system for open applications. Over the past years there has been a great development in speech synthesis technology. The intelligibility and comprehensibility of synthetic speech have reached an acceptable level [3]. However, text analysis capability, particularly the capability in processing real text is still rudimentary in a sense [4]. Due to this reason, the quality of synthetic speech degrades sharply in some open applications. Therefore,

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 200 – 210, 2006.

to improve the robustness, a TTS engine for open applications should involve a text analysis module that is capable of interpreting unrestricted texts with accuracy.

In this paper, we present a robust text analysis system for Chinese text-to-speech (TTS) synthesis. In converting a real orthographic Chinese text into a linguistic representation for speech synthesis, three problems usually arise, namely lexical disambiguation, unknown word resolution and non-standard words normalization. These problems interact and cannot be resolved separately. This paper proposes a unified framework to solve the three issues. First, a lexicon word or a continuum of non-Chinese characters of same type is defined as a morpheme, i.e. a basic unit forming a Chinese word. Based on this definition, the three problems can be unified in a single framework and reformulated as a two-pass tagging task on a sequence of morphemes. With a view to the higher requirement of efficiency of a TTS engine for some on-line or multi-thread applications, the two-pass task is performed under the framework of Hidden Markov Model (HMM). Our system consists of four main modules: (1) a pre-segmenter for sentence segmentation and morpheme segmentation; and (2) a lexicalized HMM-based chunker for unknown word identification and guessing; and (3) a HMM-based tagger for converting orthographic morphemes to their Chinese phonetic representation (viz. pinyin), given their word-formation patterns and part-of-speech information; (4) a post-processing for interpreting phonetic tags and fine-tuning pronunciation order for some special non-standard words (NSWs) if necessary. We test our system using a pinyin-notated corpus built from the Peking University corpus [5]. The results show that our system is very effective for text analysis in Chinese TTS.

The remainder of this paper is organized as follows: Section 2 introduces the three issues in Chinese text analysis. Section 3 defines the important concepts ‘morpheme’ for Chinese TTS. Section 4 details our text analysis system. Section 5 reports the evaluation results of our system on different public corpora, and the final section is our conclusion remarks on this work.

2 Three Open Issues in Chinese Text Analysis

2.1 Lexical Disambiguation

Correct lexical disambiguation is very important for a high-quality Chinese TTS system. In general, Chinese lexical ambiguities involve (1) word segmentation ambiguity, (2) part-of-speech ambiguity and (3) pronunciation ambiguity (viz. the problem of polyphonic words), which usually results in ambiguous interpretation of a given text.

(a) 南京/ns 市长/n 江/nr 大桥/nr
(Mayor of Nanjing City Jiang Daqiao)
nan2jing1 shi4zhang3 jiang1 da4qiao2
(b) 南京市/ns 长江/ns 大桥/n
(Nanjing Changjiang River Bridge)
nan2jing1shi4 chang2jiang1 da4qiao2

Fig. 1. Different interpretation of the ambiguous fragment “南京市长江大桥”

Fig. 1 illustrates different interpretations of the ambiguous fragment “南京市长江大桥”. This fragment has two meaningful segmentations: in segmentation (a), it means ‘Mayor of Nanjing City Jiang Daqiao’ and pronounces ‘nan2jing1 shi4zhang3 jiang1 da4qiao2’, while in segmentation (b), it means ‘Nanjing Changjiang River Bridge’ and the relevant pronunciation is ‘nan2jing1shi4 chang2jiang1 da4qiao2’.

2.2 Resolution of Unknown Words

Resolving unknown words with accuracy is a key challenge to a robust text analysis system for open applications. In fact, most current TTS systems involve a dictionary to specify the necessary linguistic information of words such as part-of-speech categorization and pronunciation for synthesis. However, no dictionary could be complete in practice. Although a predefined dictionary may cover most words in use, there are many other words, such as proper nouns and domain-specific terms can not be exhaustively enumerated in a dictionary. Therefore, to generate synthetic speech correctly and naturally for an open-end text in Chinese, a TTS engine should be able to resolve different types of unknown words whose linguistic information are not defined in the dictionary being used. Chinese unknown word resolution mainly involves unknown word identification (UWI), unknown word guessing (UWG) and unknown word pronunciation (UWP).

(a) 国学大师张中行去世 (from *Xinhua News Agency February 25, 2006*)
(Master of Chinese culture Zhang Zhongxing passed away)
国学/n 大师/n 张/nr 中行/nr 去世/v
guo2xue2 da4shi1 zhang1 zhong1xing2 shi4shi4

(b) 中行长葛支行注重健身 (from the *1st SIGHAN Bakeoff PK-Open test* set)
(The Changge Branch of The Bank of China put stress on gymnastic exercises)
中行/nz 长葛/ns 支行/n 注重/v 健身/vn
zhong1hang2 chang2ge3 zhi1hang2 zhu4zhong4 jian4shen1

Fig. 2. Different interpretation of the unknown word “中行” in different contexts

Fig. 2 illustrates an example of unknown words in Chinese text. The unknown word “中行” has the same form but differ in part-of-speech, meaning, and pronunciation: in sentence (a), it is a Chinese first-name pronouncing zhong1xing2. But in sentence (b), it is an abbreviation of an organization name, i.e. ‘The Bank of China’ and its pronunciation is zhong1hang2.

2.3 Normalization of Non-standard Words

Real Chinese texts are very mess and often contain a number of non-hanzi characters, which raise another key issue for Chinese TTS, namely the identification and normalization of NSWs in Chinese text. However, Chinese text normalization is by no means an easy task. On the one hand, ambiguities usually arise while pronouncing non-hanzi characters in a NSW. For example, the symbol + has four possible pronunciations in different NSWs, i.e. zheng4 ‘positive’ in a normal number like +0.05, ling2zhang4 ‘above zero’ in a temperature expression like +20℃ ‘20 degrees above zero Celsius’, jia1 ‘plus’ in a expression of arithmetic computation like 1+1=2, and a functional symbol to be skipped in some telephone number expression such as +852-12345678. On the other hand, the order of pronouncing a NSW is not always from left to right and may change in some cases. The percent 5%, for example, should be pronounced from right to left. Consequently, more information, even pragmatic information is necessary to achieve correct interpretation of NSWs in Chinese text. For example, it is very difficult to determine only by using the local contextual information whether the word 7:15 in the sentence 现在是7:15 is a score pronouncing qi1bi3shi2wu3 ‘seven to fifteen’ or a time expression pronouncing qi1dian3shi2wu3 ‘fifteen past seven’.

3 Representation of Words in Chinese Text

3.1 A Typology of Chinese Words

As shown in Table 1, words in Chinese text can be categorized into standard words and non-standard words (NSWs), based on the types of characters consisting of a word. A standard word is formed by a string of Chinese characters (viz. Hanzi) while a non-standard word contains at least one non-Chinese character like numerals, English letters, punctuation marks and other symbols. Standard words can be sub-divided into two groups: lexicon words (LWs) that are included in the lexicon being used and unknown words (UWs) that are unseen in the lexicon. NSWs can be further classified into four types: numeral expressions, English words, punctuations and other NSWs.

Table 1. Number of Chinese words in the PKU Corpus

Word type	Example
Lexicon words (LWs)	
Ambiguous in part-of-speech	工作(verb or noun), 好(adjective or verb)
Ambiguous in pronunciation	重(zhong4 or chong2), 行(xing2 or hang2)
Unknown words (UWs)	中行, 鞍钢, …
Non-standard words (NSWs)	
Numeral expressions	15.5%, 2006年, …
English words	IBM, WWW, …
Punctuations	，,。,！,…
Other NSWs	abc-123@hotmail.com, …

3.2 Definition of Morphemes

In order to handle word-level information conveniently, we take lexicon words and continuums of non-Chinese characters as the basic units or morphemes forming a word in Chinese text. For convenience, we call them lexicon word morpheme and non-hanzi morpheme respectively. Here, a continuum of non-Chinese characters is a string of consecutive non-Chinese characters with the same category. For example,

Table 2. Types of morphemes consisting of Chinese words

Morpheme type	Definition	Example
LW	lexicon words	阿, 人民, 大学, …
DIG	digits or numerals	0, 123, １２３, …
ALP	alphabets or letters	A, abc, ａｂｃ, …
PUN	punctuation marks	，，。，！, …
SYM	other symbols	@, #, $, …

the word 2006年 'the year of two thousand and six' is composed of two morphemes, a non-hanzi morpheme (viz. the continuum of consecutive digits '2006') and a lexicon word morpheme (viz.年, 'year').

As shown in Table 2, morphemes can be categorized to five major types, namely *LW* (lexicon words), *DIG* (digits or numerals), *ALP* (alphabets or letters), *PUN* (punctuation marks) and *SYM* (other symbols), based on the types of characters consisting of a morpheme.

Table 3. Phonetic tags for non-hanzi morphemes

Morpheme	Phonetic tag	Description	Example
DIG	NAD	A numeral morpheme pronouncing as a decimal integer	12.34%
	DBD	A numeral morpheme pronouncing digit by digit	12.34%
ALP	AAW	A alphabet morpheme pronouncing as a (English) word	Tel: +852-12345678
	LBL	A alphabet morpheme pronouncing letter by letter	IBM
SYM	STP	A symbol morpheme to be pronounced	12.34%
	SAF	A function symbol with no pronunciation	Tel: +852-12345678

A non-hanzi morpheme usually has different pronunciations in different contexts. For example, a numeral morpheme may pronounce either as a decimal integer or digit by digit, depending on what types of words it forms and where it is in the words. In order to unify the normalization of NSWs with the processing of standard words, non-hanzi morphemes are further classified into different sub-types in terms of the way of pronunciation. As shown in Table 3, a number of phonetic tags are defined to represent the relevant sub-types of non-hanzi morphemes.

3.3 Representation of POS-Tagged Chinese Words

In practice, any segmented word in a real Chinese text consists of one or more morphemes defined in Table 2 if the system dictionary covers all possible Hanzi (viz.

Chinese characters). In particular, a morpheme has four possible patterns to present itself after word segmentation: (1) It is an independent segmented word by itself. (2) It is an initial morpheme of a segmented word. (3) It is a mid morpheme of a segmented word. (4) It is a final morpheme of a segmented word. In this paper, we use four tags *O*, *I*, *M* and *F* to denote these patterns, respectively.

(a) 应胡锦涛主席的邀请，北朝鲜领导人金正日从4月19日至21日对中国进行了非正式访问。(North Korean leader Kim Jong-Il paid an unofficial visit to China from April 19 to 21 at the invitation of Chinese president Hu Jintao)

(b) 应/p 胡/nr 锦涛/nr主席/n 的/u 邀请/vn，北朝鲜/ns 领导人/n 金/nr 正日/nr 从/p 4月/t 19日/t 至/p 21日/t 对/p 中国/ns 进行/v 了/u 非正式/b 访问/vn 。/w

(c) 应/p-O 胡/nr-O 锦/nr-I 涛/nr-F 主席/n-O 的/u-O 邀请/vn-O，北朝鲜/ns-O 领导人/n-O 金/nr-O 正/nr-I 日/nr-F 从/p-O 4/t-I 月/t-F 19/t-I 日/t-F 至/p-O 21/t-I 日/t-F 对/p-O 中国/ns-O 进行/v-O 了/u-O 非/b-I 正式/b-F 访问/vn-O 。/w-O

Fig. 3. An example of representing a POS-tagged sentence as a sequence of tagged morphemes. (a) a plain Chinese sentence. (b) a POS-tagged sentence. (c) a sequence of tagged morphemes.

With these word-formation tags, a POS-tagged sentence can be equivalently represented as a sequence of morphemes attached with their relevant hybrid tags (as illustrated in Fig. 3). A hybrid tag has the format: T1-T2, where T1 denotes a POS tag and T2 denotes a word-formation tag.

4 The Unified Framework for Chinese Text Analysis

4.1 Overview

As shown in Fig. 4, our system interprets a text in four main steps: Firstly, a pre-segmenter is used to segment a given plain text in Chinese into sentences and further segment each sentence to morphemes by using lexicon word bigrams [6]. Secondly, a lexicalized chunker is applied to assign each morpheme a proper hybrid tag defined in

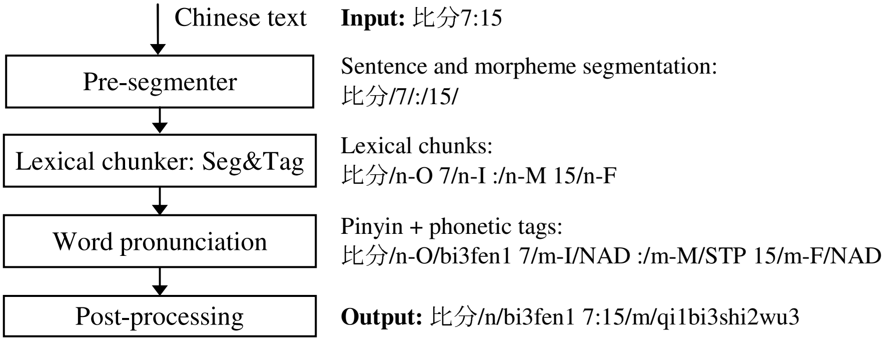

Chinese text
Input: 比分7:15
Pre-segmenter
Sentence and morpheme segmentation:
比分/7/:/15/
Lexical chunker: Seg&Tag
Lexical chunks:
比分/n-O 7/n-I :/n-M 15/n-F
Word pronunciation
Pinyin + phonetic tags:
比分/n-O/bi3fen1 7/m-I/NAD :/m-M/STP 15/m-F/NAD
Post-processing
Output: 比分/n/bi3fen1 7:15/m/qi1bi3shi2wu3

Fig. 4. Overview of the text analysis system for Chinese TTS

Section 3.3. Thirdly, a word pronunciation module is to notate each LW morpheme with a proper pinyin string and each non-hanzi morpheme with a proper phonetic tag defined in Table. 3. Finally in the forth step, a set of pattern rules [8] are used to interpret the phonetic tags and fine-tune pronunciation order for some special NSWs like percentages and fractions. At the same time, the tagged and notated morphemes are merged to words according to their respective word-formation tags. The following two sections will detail lexical chunking and word pronunciation, respectively.

4.2 Lexical Chunking

Based on the reformulation of a POS-tagged word as a sequence of tagged morphemes, word segmentation and POS tagging can be unified as a lexical chunking task on a sequence of morphemes. In order to keep balance between accuracy and efficiency, a uniformly lexicalized HMM-based chunker [7] is applied in our system.

Given a sequence of morphemes $M = m_1 m_2 \cdots m_n$, the lexicalized HMM-based chunker aims to find an appropriate sequence of hybrid tags $\hat{T} = t_1 t_2 \cdots t_n$ that maximizes the following score

$$\hat{T} = P(T \mid W) = \arg\max_{T} \prod_{i=1}^{n} P(m_i \mid m_{i-N,i-1}, t_{i-N,i}) P(t_i \mid m_{i-N,i-1}, t_{i-N,i-1}) \qquad (1)$$

Equation (1) presents an N-order lexicalized HMMs for lexical chunking. In comparison with standard HMMs, lexicalized HMMs can handle both contextual morphemes and contextual tags for the assignment of hybrid tags to morphemes, which will result in an improvement of precision. In view of serious data sparseness in higher-order models, we employ the first order lexicalized HMMs in our system.

4.3 Word Pronunciation

Once lexical chunking is done, the next task is to find correct pronunciations for the respective tagged morphemes. Word pronunciation performance exerts a direct influence on the correctness of the resulting synthetic speech. However, high-accuracy word pronunciation for Chinese is a challenge because Chinese writing system is not phonetically transparent and many words in Chinese text are ambiguous in pronunciation. Most previous work applied rules to resolve pronunciation ambiguity [9] [10]. However, a large-coverage set of rules for disambiguation is usually difficult to acquire. In this study, we take word pronunciation as a tagging problem on a sequence of tagged morphemes and propose a HMM-based tagger to notate a LW morpheme with a proper pinyin string or label a non-hanzi morpheme with a phonetic tag shown in Table 3, in which the word-formation patterns and part-of-speech tags yield in lexical chunking are combined for word pronunciation disambiguation (WPD) and UWP.

In fact, lexical information, particularly POS information plays an important role in disambiguating Chinese polyphonic words. In practice, a corresponding relationship may exist between part-of-speech and pronunciation for most polyphonic words in Chinese. Our survey on the lexicon being used shows that among a total of 453 polyphonic words under discussion, 404 polyphonic words have different pronunciations corresponding to different part-of-speech categories. In other words, about 90% of polyphonic words have a one-to-one mapping between their POS categories and pronunciations. According to our further investigation on the PKU corpus [5], over 89% polyphonic words can be completely or partly resolved if their part-of-speech categories are correctly given.

The HMM-tagger performs word pronunciation on a sequence of tagged morphemes in two main steps: (1) The first step generates a set of pronunciation candidates for each morpheme in the input by consulting the lexicon being used. In this step, some polyphonic morphemes are (partly) resolved by filtering the ineligible candidates with their part-of-speech categories and word-formation patterns; (2) In the second step, the first-order HMMs shown in Equation (2) are used to score the rest pronunciation candidates and find a proper one for each morpheme.

$$\hat{Y} = \arg\max_{Y} \prod_{i=1}^{n} P(m_i \mid y_i) P(y_i \mid y_{i-1}) \qquad (2)$$

Where, $m_i (1 \le i \le n)$ denotes a morpheme in the input and $y_i (1 \le i \le n)$ denotes a pinyin candidate of m_i.

5 Experimental Results and Discussion

5.1 Experimental Data

In evaluating our system, we conduct a number of experiments on the Peking University (PKU) corpus. The original PKU corpus contains six month of news texts from *the People's Daily* (January to June in 1998), and is manually segmented and tagged with part-of-speech by the Peking University [5]. In order to train the models for word pronunciation, we notated this corpus with pinyin. In this study, the first month is used for testing while the other five months of data are for training. It should be noted that word segmentation, POS tagging and word pronunciation are evaluated separately in our experiment, although the three tasks are performed in a unified framework in our system.

Table 4. Number of words in the PKU corpus

Corpus	LW			UW	NSW	Total
	AmbPos	AmbPy	Total			
Test (Jan)	491K	161K	866K	56K	197K	1,120K
Training (Feb-Jun)	2,721K	892K	4,785K	314K	1,065K	6,166K
Total	3,212K	1,053K	5,651K	370K	1,262K	7,286K

As illustrated in Table 4, there are a total of 7,286K words in the PKU corpus, among which 5,651K (viz. 77.56%), 370K (viz. 5.08%) and 1,262K (viz. 17.32%) are lexicon words, unknown words and non-standard words, respectively. Furthermore, about 56.84% and 18.63% lexicon words are observed to be ambiguous in part-of-speech and pronunciation.

In addition to the PKU corpus, a lexicon of about 65K entries is used, which are mainly from the Modern Chinese Grammar Information Lexicon of the Peking University [5]. In order to make this lexicon complete, a number of GBK Hanzi are also added to the lexicon.

5.2 Experimental Results

Table 5 presents the evaluation results for word segmentation using the PKU corpus. In this evaluation, the standard HMMs and the view of taking character as the basic unit forming Chinese words (viz. the character-based morpheme) are also introduced as the baseline. Table 5 reveals a number of observations. Firstly, the lexicalized HMMs consistently perform better over the corresponding standard HMMs for all experimental conditions, in particular for character-based morphemes. As shown in Table 5, the lexicalization technique can improve the overall segmentation F-measure by 1.3 percent for the morphemes defined in Table 2, while the number is 8.3 percent for character-based morphemes. Secondly, systems using the morphemes defined in Table 2 outperform those taking characters as morphemes, particularly in case of standard HMMs.

Table 5. Evaluation results for word segmentation using the PKU corpus

Morpheme	Measure	Standard HMMs				Lexicalized HMMs			
		Overall	LW	UW	NSW	Overall	LW	UW	NSW
Character-based morphemes	F	88.2	89.0	53.3	98.1	96.5	96.9	81.8	99.4
	R	88.1	87.0	70.3	98.0	96.4	96.4	86.1	99.3
	P	88.3	91.1	42.9	98.3	96.7	97.4	77.9	99.5
The morphemes defined in Table 2	F	95.7	96.2	76.6	99.0	97.0	97.2	86.0	99.4
	R	95.4	95.7	78.1	98.8	97.2	97.5	85.6	99.3
	P	96.0	96.8	75.2	99.1	96.8	96.9	86.4	99.5

Table 6. Evaluation results for word segmentation using the SIGHAN Bakeoff PK-open data

Test track	R_{OOV}	R_{iv}	R	P	F	$F_{bakeoff-best}$
1st Bakeoff PK-Open	83.8	97.0	96.6	96.1	96.3	95.9
2nd Bakeoff PK-Open	83.9	96.1	96.2	96.4	96.3	96.9

We also test our system using the first and second SIGHAN Bakeoff PK-Open data [11][12]. The respective best F-measures for the two tracks are 95.9% and 96.9%. As can be seen from Table 6, our system yields an F-measure of 96.3% for the two tracks. This demonstrates in a sense that our system can achieve state-of-the-art performance in word segmentation.

Table 7 presents the evaluation results of our system for POS tagging using the PKU corpus. In this evaluation, the input is a segmented text and the standard tagging accuracy is computed to evaluate the tagging performance of the system. The results show that the introduction of lexicalization technique helps improve tagging accuracy. In comparison with the standard HMMs, the lexicalized HMMs can improve the

Table 7. Evaluation results for POS tagging

Method	Overall	AmbLW	UW	NSW
Standard HMMs	94.3	89.8	86.4	99.9
Lexicalized HMMs	96.1	93.2	90.2	99.9

tagging accuracy respectively by 1.8 percent for all words, 3.4 percent for ambiguous lexicon words and 3.8 percent for unknown words.

Table 8 presents the evaluation results for word pronunciation. The input of this evaluation is a segmented and POS-tagged text, which will be converted to a text of tagged morphemes shown in Fig. 3 before word pronunciation. Furthermore, the baseline method, namely the dictionary-based disambiguation is also involved for, which disambiguates polyphonic LWs by consulting the dictionary with their POS tags or performs unknown word pronunciation using POS and word-formation patterns. As can be seen from Table 8, our system improves the pronunciation accuracy of polyphonic LWs from 94.7% to 97.2 and the pronunciation accuracy of UWs from 96.2% to 98.1%, in comparison with the baseline method. Furthermore, our system yields an accuracy of 98.5% for NSW pronunciation.

Table 8. Evaluation results for word pronunciation

	Polyphonic LW	Unknown word	NSW
Dictionary-based method	94.7	96.2	-
Our system	97.2	98.1	98.5

6 Conclusion

This paper presents a robust text analysis system for Chinese TTS, which consists of four main components: (1) a pre-segmenter for sentence segmentation and morpheme segmentation; and (2) a lexicalized HMM-based chunker for unknown word identification and guessing; and (3) a HMM-based tagger for word pronunciation; (4) a post-processing for interpreting phonetic tags and fine-tuning pronunciation order for some special NSWs if necessary. In this system, the three key issues in Chinese text analysis, namely lexical disambiguation, unknown word resolution and non-standard word normalization are solved in a unified framework of Hidden Markov Models (HMMs). As a result, the accuracy of the system can be improved without losing its efficiency. We test our system on a pinyin-notated corpus built from the PKU corpus and the SIGHAN Bakeoff data. The experimental results demonstrate the effectiveness of the proposed unified framework. In this study, we focus our work on processing texts in simplified Chinese. For future work, we would like to apply our current method to Taiwan Mandarin and Cantonese TTS.

Acknowledgments. We would like to thank the Institute of Computational Linguistics at Peking University for their corpus and lexicon.

References

1. Shih, Chilin, and Richard Sproat: Issues in text-to-speech conversion for Mandarin, Computational Linguistics and Chinese Language Processing, Vol.1, No.1 (1996) 37 -86
2. Xu, Jun, Guohong Fu, and Haizhou Li: Grapheme-to-Pinyin for Chinese text-to-speech system, In: Proceedings of the 8th International Conference on Spoken Language Processing (INTERSPEECH 2004 - ICSLP), Jeju Island, Korea (2004) 1885-1888
3. Lemmetty, Sami: Review of speech synthesis technology, Master's Thesis, Helsinki University of Technology, Finland, 1999
4. Sproat, Richard, Alan Black, Stanley Chen, Shankar Kumar, Mari Ostendorf, and Christopher Richards: Normalization of non-standard words. Computer Speech and Language, Vol.15, No. 3 (2001) 287-333
5. Yu, Shiwen, Houfeng Duan, Xuefeng Zhu, Bin Swen, and Baobao Chang: Specification for corpus processing at Peking University: Word segmentation, POS tagging and phonetic notation. Journal of Chinese Language and Computing, Vol.13, No.2, (2003) 121-158
6. Fu, Guohong, and Kang-Kwong Luke: Chinese unknown word identification using class-based LM. Lecture Notes in Artificial Intelligence (IJCNLP 2004), Vol. 3248 (2005) 704-713
7. Fu, Guohong, and Kang-Kwong Luke: Chinese named entity recognition using lexicalized HMMs. ACM SIGKDD Explorations Newsletter, Vol.7, No.1 (2005) 19-25
8. Fu, Guohong: User rule specification for text normalization. InfoTalk Technical Report, InfoTalk -R&D -2002-001
9. Zhang, Zirong, Min Chu and Eric Chang: An efficient way to learn rules for grapheme-to-phoneme conversion in Chinese, In: Proceedings of 2002 International Symposium on Chinese Spoken Language Processing (ISCSLP 2002), Taipei, Taiwan (2002) 59-63
10. Zheng, Min, Qin Shi, Wei Zhang, and Lianhong Cai: Grapheme-to-phoneme conversion based on TBL algorithm in Mandarin TTS system. In: Proceedings of ITERSPEECH 2005, Lisbon, Portugal (2005) 1897-1900
11. Sproat, Richard, and Thomas Emerson: The first international Chinese word segmentation bakeoff. In: Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, Sapporo, Japan (2003) 133-143
12. Emerson, Thomas: The second international Chinese word segmentation bakeoff. In: Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing, Jeju Island, Korea (2005) 123-133

Speech Synthesis Based on a Physiological Articulatory Model

Qiang Fang[1,2] and Jianwu Dang[1]

[1] IIPL, school of information science, Japan Advance Institute of Science and Technology
[2] Phonetics Lab., Institute of Linguistics, Chinese Academy of Social Sciences
jdang@jaist.ac.jp, fq0237@yahoo.com.cn

Abstract. In this paper, a framework for speech synthesis is proposed to realize the process of speech production of human, which is based on a physiological articulatory model. Within this framework, it begins with given articulatory targets, then muscle activation patterns are estimated according to the targets by accounting for both the equilibrium characteristics and muscle dynamics, consequently, the articulatory model is driven to generate a time-varying vocal tract shape corresponding to the targets by contracting the corresponding muscles. Thereafter, a transmission line model is implemented for the time-varying vocal tract to produce speech sound. At last, a primary experiment is carried out to synthesize the single vowels and diphthongs of Chinese with the physiological articulatory model based synthesizer. The result shows that the spectra of the synthetic sound for single vowels are consistent with those of the real speech, and proper acoustic characteristics are obtained in most cases for diphthongs.

Keywords: physiological articulatory model, speech production, acoustic model, speech synthesis, Chinese vowel.

1 Introduction

The first synthesizer was constructed by Kratzenstein in 1779, which was a mechanical model composed of the vocal tract, glottal and lung. Since then, speech synthesis has gone through the stages: mechanical machine, circuit based method, electronic based facilities, and ultimately computer based algorithms.

Before 1990's, the mainstream of speech synthesis was formant based synthesis, which requires small memory and less computation resource, because of the limitation of computer power. With the development of computer science and speech technology, concatenative synthesis based on large-scale corpus becomes popular due to its priority in synthesizing fairly intelligible and natural speech sounds. However, this kind of synthesizer heavily depends on the prerecorded speech corpus and lacks flexibility to generate various styles of speech, especially emotional and personalized speech, with high quality.

One alternative to solve these kinds of problems is articulatory based synthesis, which generates speech sounds by imitating the mechanisms of speech production of

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 211 – 222, 2006.

human. In this paper, a framework of articulatory synthesis is presented based on a physiological model.

2 Physiological Model

A partial 3D model with a thick sagittal layer of the tongue has been constructed based on volumetric MR images using an extended finite element method (X-FEM), where the MR images are obtained form a male Japanese speaker. The outlines of the tongue are extracted from two sagittal slices: one is the midsagittal plane and the other is a plane 1.0cm apart from the midsagittal on the left side. The outline of left side is duplicated to the right side with an assumption that the left and right sides of the tongue are symmetrical.

Mesh segmentation of the tongue tissue roughly copy the fiber orientation of the genioglossus. The outline in each sagittal plane is divided into 10 radial sections which fan out from the attachment of the genioglossus on the jaw to the tongue surface. While in perpendicular direction, the tongue is divided into 6 sections. Eventually, a 3D mesh model is built by connecting the intersection nodes in the midsagittal plane to corresponding nodes on the left and right side, accordingly, each mesh is a brick with 8 corners. Fig. 1 illustrates the tongue model based on this segmentation. Ultimately, the tongue tissue is represented as 120 hexahedrons, each hexahedron is modeled by 28 viscoelastic solid cylinders (12 edges, 2 cross-wise connection in each surface of the hexahedron, and 4 connections between 8 diagonal vertices inside the hexahedron), which have not only masses but also volumes.

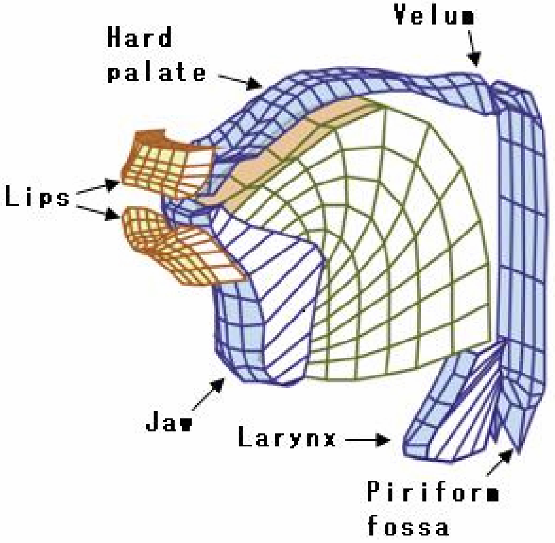


Fig. 1. The oblique view of the physiological articulatory model

To generate the shape of the vocal tract, an articulatory model should include the lips, teeth, tongue, hard palate, soft palate, pharyngeal wall and larynx. At present, the lips and soft palate are taken into account when constructing area function for the vocal tract, though they are not modeled physiologically.

Outlines of the vocal tract wall and mandibular symphysis are extracted from MRI images in the midsagittal and parasagittal planes (0.7 and 1.4cm form the midsagittal plane one the left side), then copy the configuration of the left side to the right side. The model for the articulators is shown in Fig. 1 [2].

3 Control Mechanism

The extrinsic muscles (genioglossus, geniohyoid, hyoglossus, styloglossus) and intrinsic muscles (superior longitudinalis, inferior longitudinalis, transversus, and verticalis) of the tongue, as well as the rigid organs (jaw and hyoid bone), are taken into account for manipulating the motion of the articulatory model.

To drive the physiological articulatory model, a target-based strategy has been developed. It consists of two parts: one is muscle workspace [3], and the other is an equilibrium position mapping (EP-map) [2]. Several representative points are chosen to represent the motion of the model, namely control points, which are used to control the shape and/or position of the tongue and the jaw. They are the apex of the tongue in the midsagittal plane for the tongue tip, a weighted average position of the highest three points of the midsagittal plane in the vocalic configuration for the tongue dorsum, a point 0.5cm inferior to the tip of the mandible incisor for the jaw.

The EP-map associates muscle forces with the equilibrium position of control points, in spite of where the start position is. That's to say, if a certain force is given to a specific muscles, the control points are bound to converge to their equilibrium positions no matter where they start from (see [2] for the details). It reflects a static force for control of the model. Fig.2 shows the EP-map for the tongue tip and tongue dorsum.

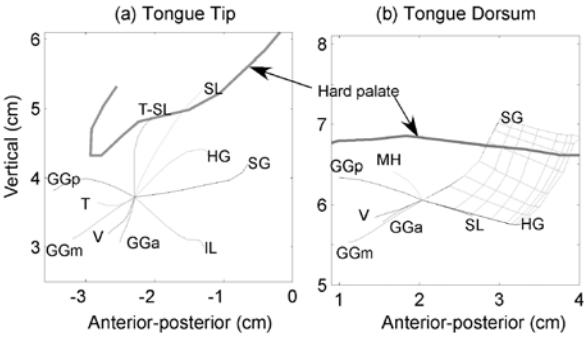
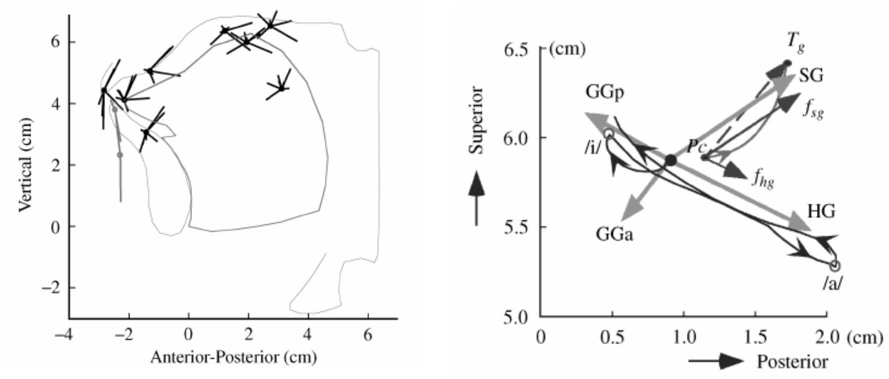


Fig. 2. EP-maps of the muscle activation and articulator location: (a) EP-map for tongue tip (b) EP-map for tongue dorsum. The curves spreading out from the central point are the trajectory of the equilibrium position for the control points as the activation force increases from 0 to 6 N.

The muscle workspace is a description of the relationship between the muscle activation and the displacement of control points of the articulators. It accounts for the dynamic characteristics of articulation by reducing the distance between the current position and target for each control point with the stepwise method. At first, four typical muscle workspaces are set up for both the tongue tip and tongue dorsum respectively, and two muscle workspaces for the jaw, as shown in the left panel of Fig.3. Then, a dynamical muscle workspace for the current position is derived by a nonlinear interpolation based on the typical muscle workspaces. When projecting the articulatory vector of current point (*Pc*) to the target (*Tg*) onto the dynamic muscle workspace, a force projection is generated for each muscle. Only the projection that positively correlates to the articulatory vector is taken into account for the control (see [3], [4] for the details).



(a) (b)

Fig. 3. Muscle workspace. (a)Typical muscle workspaces for the tongue tip, tongue dorsum and jaw (b) an example for estimation of force vector based on the muscle workspace.

4 Underlying Acoustic Model

By far, the physiological model and its control strategy have been briefly introduced. For given targets, the articulators are driven to the desired position by appropriate muscle contraction under the control of the strategy, and the shape of the vocal tract is formed by the surfaces of the articulators.

In order to facilitate estimating the acoustic features of the vocal tract, a gridline system is adopted to describe the width of the vocal tract in the midsagittal and parasagittal plane (as shown in the left panel of Fig. 4, gridline system consists of the thin lines), which are used to estimate the area function with the improved $\alpha-\beta$ model (see [5] for the details). Ultimately, the vocal tract is divided into 30 sections according to the representation based on the gridline system. As for the nasal cavity, it is divided into 12 sections. Most of these sections have constant cross-sectional areas, except the sections around the nasal-pharyngeal port. And for each section, both in the nasal cavity and in the vocal tract, a transmission line model is adopted to simulate its characteristics, which is described by the following equations:

$$(P_r - P_j)A_j + \rho_0 A_j \frac{x_j}{2}\frac{\partial v}{\partial t} + rS_j \frac{x_j}{2} v = 0 \tag{1}$$

$$(U_{j+1} - U_j)\rho_0 \Delta t + [(\rho_0 + \Delta\rho_j)(A_j + \Delta A_j) - \rho_0 A_j]x_j = 0 \tag{2}$$

$$\frac{PV^\gamma}{T} = const \tag{3}$$

$$m_j \frac{\partial^2 y}{\partial t^2} + b_j \frac{\partial y}{\partial t} + k_j y = PS_j \tag{4}$$

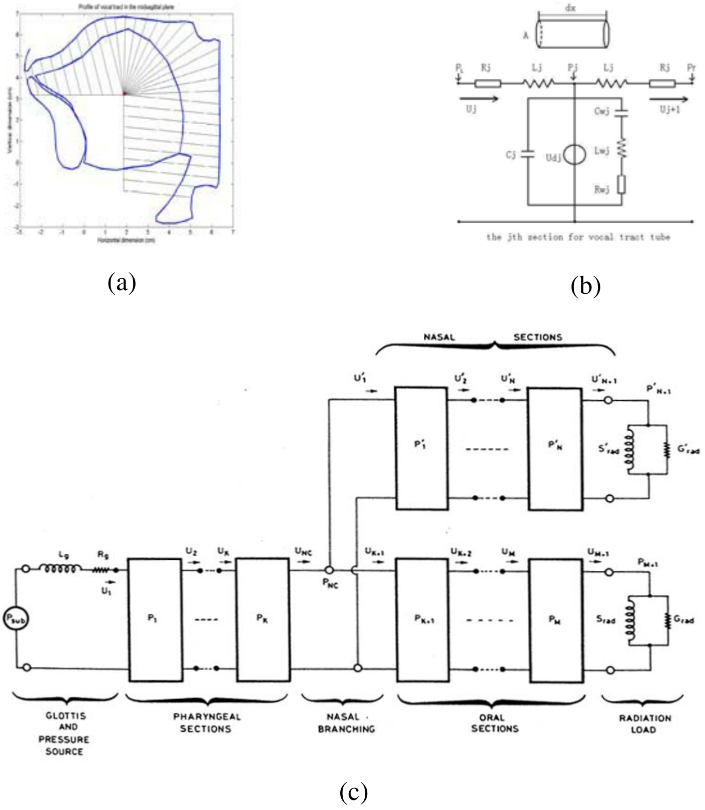
$$A_j = A_{0j} + yS_j \tag{5}$$

Where P_j , P_r are the pressures at the middle and right end of the jth sub-tube respectively, A_j is the cross-sectional area of the jth sub-tube, A_{0j} is the cross-sectional area of the jth sub-tube when the vocal tract wall is at its equilibrium position, v is the velocity of air particles within the jth sub-tube, S_j is the perimeter of the jth sub-tube, ρ_0 is the air density at the equilibrium state, U_{j+1} and U_j are volume velocities, and have the relationship with v_j and v_{j+1} : $U_j = v_j A_j$, $U_{j+1} = v_{j+1}A_{j+1}$, where v_j and v_{j+1} are the particle velocity at the inlet and outlet of the jth sub-tube respectively; m_j , b_j and k_j are mass, viscosity, and mechanic capacity of the wall per unit length of the jth sub-tube respectively, and y is the displacement of the vocal tract wall. Here, equation 1 reveals the relationship based on Newton second law, equation 2 reflects the law of mass conservations, equation 3 is the gas law, and equation 4 discloses the phenomenon of wall vibration. To simplify equation 1 and 2, the volume velocity U within the jth section is represented by U_j for the left part and U_{j+1} for the right part, while the pressure P within the sub-tube is represented by the pressure, P_j , at the middle of the sub-tube. Eventually, the following equations are derived for the transmission line model of a single uniform sub-tube:

$$P_j - P_r = \frac{\rho_0 x_j}{2A_j}\frac{\partial U_{j+1}}{\partial t} + \frac{rS_j x_j}{2A_j^2}U_{j+1} \tag{6}$$

$$U_j - U_{j+1} = \frac{A_j x_j}{\rho_0 c^2}\frac{\partial P_j}{\partial t} + x_j \frac{\partial A_{0j}}{\partial t} + x_j S_j \frac{\partial y}{\partial t} \tag{7}$$

Let $L_j = \dfrac{\rho_0 x_j}{2A_j}$, $R_j = \dfrac{rS_j x_j}{2A_j^{\ 2}}$, $C_j = \dfrac{A_j x_j}{\rho_0 c^2}$, $L_{wj} = \dfrac{m_j}{x_j S_j^{\ 2}}$, $R_{wj} = \dfrac{b_j}{x_j S_j^{\ 2}}$, $C_{wj} = \dfrac{k_j}{x_j S_j^{\ 2}}$, $U_{dj} = x_j \dfrac{\partial A_{0j}}{\partial t}$, then the equivalent circuit unit is built in the right panel of Fig. 4. Therefore, a transmission line model for the supra-glottal system is obtained by cascading all the sub-tubes (as shown in the bottom panel of Fig. 4). The branch for the nasal cavity only exists in producing nasal sounds. The details for calculating the volume velocity and pressure in each sub-tube and the performance of the acoustic system are described in [9].



(a)

(b)

(c)

Fig. 4. The supra-glottal system and its transmission line model. (a) The profile of the vocal tract for producing sound /ə/. The thin lines set up the gridline system, which partition the vocal tract into 3 major parts: polar system part, horizontal part and vertical parts. (b) Transmission line model for a sub-tube (c) Transmission line model for supraglottal system (adopted from [9]).

As for the piriform fossa, a side branch behind the larynx, and the nasal sinuses, whose details were reported in [6], [7], and [8], they are modeled as Helmholtz resonators. For a Helmholtz resonator, set the cross-sectional area of the neck is A , the length of the neck is l , and the volume of container is V , the following equations are derived:

$$F = P_{in}A - A^2 \frac{dP}{dV} x - Rv \tag{8}$$

$$\frac{dP}{dV} = -\frac{rP_0}{V} \tag{9}$$

According to the Newton second law, the following equation is formulated:

$$\rho l A \frac{d^2 x}{\partial t^2} + \frac{R}{A}\frac{dx}{\partial t} + \frac{A^2 r P_0}{V} x = P_{in} A \tag{10}$$

Where r is the heat capacity ratio, ρ is the air density, R is the viscous resistance caused by the wall of the neck, P_{in} is the pressure at the inlet of Helmholtz resonator, P_0 is the undisturbed pressure inside the Helmholtz resonator, and x is the displacement of the air column within the neck. Let $U = A\frac{dx}{dt}$, then $x = \int \frac{U}{A} dt$, $\frac{d^2x}{dt^2} = \frac{1}{A}\frac{dU}{dt}$, hence, a new equation is generated:

$$\rho l \frac{dU}{dt} + \frac{R}{A^2} U + \frac{ArP_0}{V} \int U dt = P_{in} \tag{11}$$

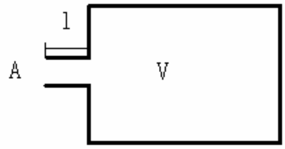


Fig. 5. Helmholtz resonator. l is the length of the neck, A is the cross-sectional area of neck, V is volume of Helmholtz resonator.

At the glottis, a glottal waveform model is used to generate the sound source for voiced sound. Nevertheless, a noise source is generated at the constriction along the vocal tract for the voiceless noise (turbulence) [9].

5 Speech Synthesis

In above sections, each part of the physiological articulatory model based speech synthesizer has been described individually. In this section, the flowchart of speech synthesis is given systematically, and synthesis experiments are carried out on Chinese single vowels and diphthongs.

5.1 Flowchart for Synthesis

To produce a specific speech sound, speakers should have a set of targets, e.g. articulatory targets, and move the articulators by activating certain muscles according to the targets to generate a specific vocal tract shape, and stimulate the vocal tract with proper sources simultaneously. That's the process of speech production of human. Since the purpose of this study is to generate speech sound by simulating human's mechanism, the speech synthesizer has the potential to realize this procedure. Figure 6 gives the flowchart of the processes involved in the proposed speech synthesizer. First, the articulator targets of the control points as well as the parameters for the lip tube and source are set according to the properties of phonemes, where the latter ones are used in calculating the acoustic characteristics. Then, the static forces are estimated by the EP-map at the beginning and exploited to activate the muscles, whereas the dynamic forces are calculated based on the muscle workspace stepwise during the articulatory movement. As a result, a time-varying vocal tract is obtained, by extracting the outlines of the articulators and the side branch of nasal cavity (if nasals are planed). An acoustical model is constructed from calculating the area function and adopting the transmission line model. Speech sounds are generated by applying a subglottal pressure to the acoustical model. In this study, a sub-glottal pressure with 8cm H_2O is employed.

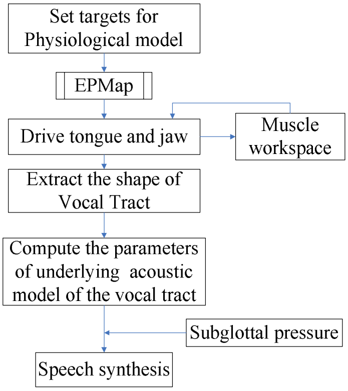

Set targets for Physiological model
EPMap
Drive tongue and jaw
Muscle workspace
Extract the shape of Vocal Tract
Compute the parameters of underlying acoustic model of the vocal tract
Subglottal pressure
Speech synthesis

Fig. 6. Flowchart for speech synthesis by applying the physiological acoustic model

5.2 Synthesis of Chinese Vowels and Diphthongs

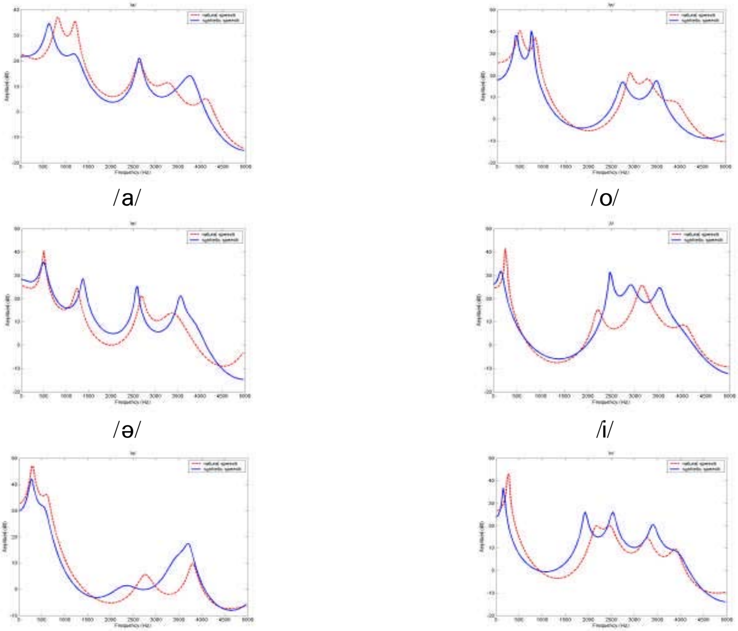
In this section, we attempt to synthesize vowels and diphthongs of Chinese with the proposed synthesizer. To do so, the first step is to define a target set for the basic elements:

vowels and consonants At present, only vowels (/a/, /o/, /ə/, /i/, /u/, /y/) and diphthongs (/ai/, /ɑu/, /ei/, /əu/, /ia/, /iɛ/, /uɑ/, /uo/, /yɛ/) are taken into account [10].

This physiological articulatory model was derived from a Japanese speaker. To obtain the targets for Chinese vowels, the difference between Japanese vowels and their corresponding Chinese vowels was investigated in the articulation level. For

Table 1. The articulatory targets for Chinese vowels (/a/, /o/, /ə/, /i/, /u/, /y/). Tt and Td represent tongue tip and tongue dorsum respectively, where the origin is at the apex of the upper incisor. (Unit: cm)

	Jaw x	Jaw y	Tt x	Tt y	Td x	Td y
/a/	0.7728	-1.5782	1.5428	-1.5282	6.3428	0.7318
/o/	0.6328	-1.3182	2.4128	-0.6582	6.7428	1.2618
/ə/	0.3828	-0.5182	1.1228	-0.5682	5.6128	1.2818
/i/	0.3828	-0.4582	1.0828	-0.7182	4.8328	2.1218
/u/	0.4528	-0.7382	1.8728	-0.3182	7.8728	1.6818
/y/	0.4028	-0.4882	1.1128	-0.4782	4.3728	2.1218

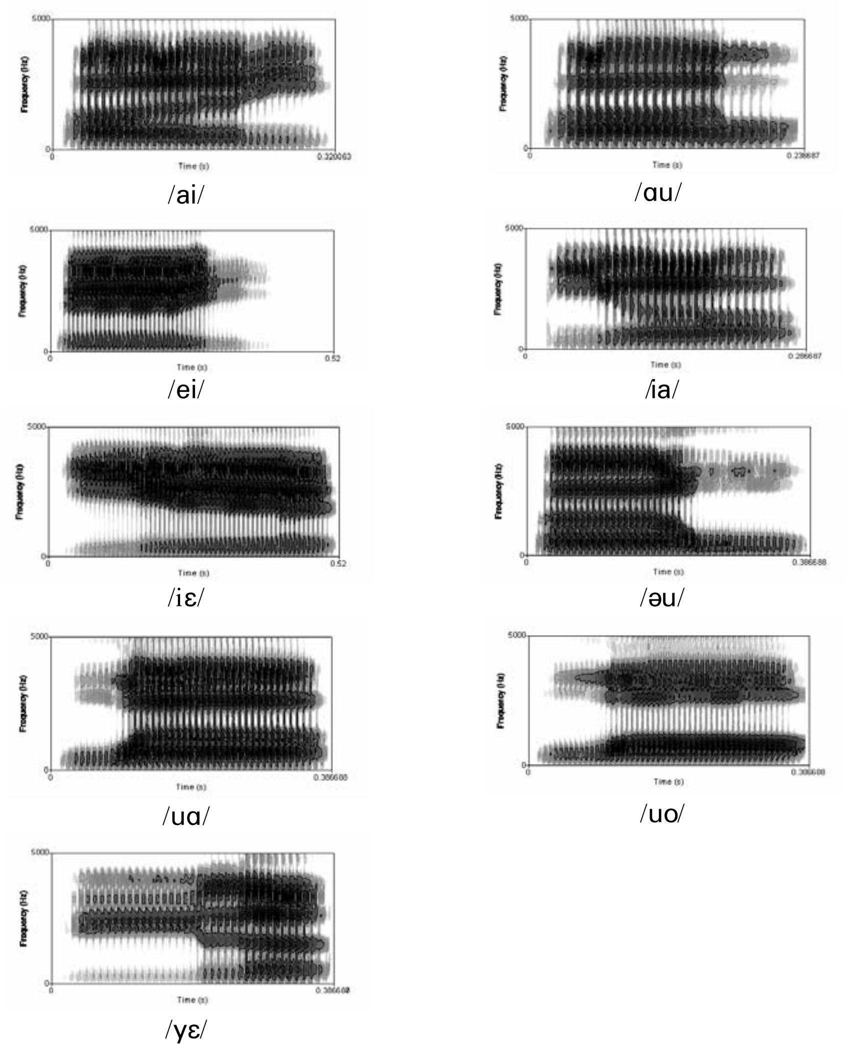


/a/ /o/

/ə/ /i/

/u/ /y/

Fig. 7. LPC based spectra for single vowels of Chinese (/a/, /o/, /ə/, /i/, /u/, /y/). The solid line represents the spectrum envelope of synthetic speech and the dash line represents that of real speech sounds (the order is 16 for LPC analysis).



/ai/ /ɑu/

/ei/ /ia/

/iɛ/ /əu/

/uɑ/ /uo/

/yɛ/

Fig. 8. Spectrogram for diphthongs of Chinese (/ai/, /ɑu/, /ei/, /əu/, /ia/, /iɛ/, /uɑ/, /uo/, /yɛ/).

Japanese vowels /a/, /o/ and /i/, they are almost the same as the corresponding vowels of Chinese. But for Chinese vowels /u/, the lip protrudes and tongue moves more backward, which result in different positions for the articulators. For /e/, the corresponding Chinese vowel is /ə/, which has a more neutral position, with the profile of the vocal tract looking like somehow a uniform tube. There is no corresponding vowel in Japanese for Chinese vowel /y/. However, the articulatory targets for this vowel can be derived from Chinese vowel /i/ by protruding the lips and moving the highest point of tongue forward.

After examining the difference between the Chinese vowels and their corresponding Japanese vowels at the articulator level, the targets for Chinese vowels are estimated based on the targets of Japanese vowels by means of analysis-by-synthesis method manually. The targets for Chinese vowels are listed in Table 1, where the origin is at the apex of the upper incisor. Fig. 7 gives the spectra of the synthetic vowels and that of real speech, which are calculated by means of LPC. It demonstrates that the spectra of the synthetic vowels are consistent with those of the real speech.

As for the diphthongs, the targets are derived from those of the single vowels. The circumstance that coarticulation occurs between the vowels, which constitute the diphthongs, is taken into account. Moreover, for Chinese, the coarticulations between vowels are always not symmetrical because one of the vowels should be more dominant than others in triphthong and diphthongs of Chinese. Therefore, there is a requirement for quantifying the degree of coarticulation for each vowel within triphthong and diphthong, which is reflected by the deviation from its typical target. The targets for the diphthongs are generated based on the above considerations. Fig. 8 gives the spectrogram of synthesized diphthongs.

6 Summary

The goal of this study is to construct a corpus independent speech synthesizer that can faithfully realize the mechanism of speech production, so that it can potentially provide a way to synthesize speech sounds with a variety of styles. In this paper, a physiological articulatory model based speech synthesizer is proposed to synthesize single vowels and diphthongs of Chinese.

As mentioned above, the physiological articulatory model is aimed to realize human's processes of speech production. For given articulatory targets, the muscle activation patterns are estimated by EP-map and muscle workspace, and employed to drive the articulators to their targets. In this way, a time-varying vocal tract is generated and its area function is estimated from the width of the vocal tract in sagittal planes. Eventually, the sound is produced by implementing the transmission line model with a proper sound source.

For a primary examination, this framework is employed to synthesize Chinese vowels and diphthongs. The results, for the single vowels, illustrate the synthetic sound have consistent spectra with real speech sound. For the diphthongs, most of them show proper characteristics in spectrogram. However, for some diphthongs, such as /Èu/ and /yQ/, there seems to be some problems with both transitions and the duration for individual phoneme.

These problems can be caused by a number of factors such as the given target, the coarticulation between the vowels, and the control strategy of the articulatory model. In the future, we will clarify the causes using MRI system and the electromagnetic articulography and improve our speech synthesizer.

Acknowledgements

This research is conducted as a program for the "21st Century COE Program" for promoting Science and Technology by Ministry of Education, Culture, Sports, Science and Technology. This study is also supported in part by the MSRA/IJARC project and in part by Grant-in-Aid for Scientific Research of Japan (No. 17300182).

References

1. Gonghuan Du, Zhemin Zhu, Xiufen Gong. : Foundation for Acoustics. Nanjing University Publishing House 2nd Edition. (2001)
2. Dang, J., and Honda, K. Construction and control of a physiological articulatory model. J. Acoust. Soc. Am. 115(2), 2004, 853–870
3. Dang, J., and Honda, K. Estimation of vocal tract shape from sounds via a physiological articulatory model. J. Phonetics, Vol30(2002), 511-532
4. Dang, J., and Honda, K. A physiological model of a dynamic vocal tract for speech production. J. Acoust. Soc. Jpn (E), Vol22(2001), 415-425
5. Dang, J., and Honda, K. Speech production of vowel sequences using a physiological articulatory model. ISCLP1998,
6. Dang, J., and Honda, K. Acoustic characteristics of the piriform fossa in models and humans. J. Acoust. Soc. Am. 101(1997), 456-465.
7. Dang, J., and Honda, K. Acoustic characteristics of the human paranasal sinuses derived from transmission characteristic measurement and morphological observation. J. Acoust. Soc. Am. 100(1996), 3374-3383.
8. Dang, J., Honda, K. and Suzuki, H. Morphological and acoustical analysis of the nasal and the paranasal cavities. J. Acoust. Soc. Am. 96(1994), 2088-2100.
9. Maeda, S. A digital simulation method of the vocal tract system. Speech Communication (1982) 199-229.
10. Wu, Z., Lin, M. Outline of experimental phonetics. Higher education Press, 1988.

An HMM-Based Mandarin Chinese Text-To-Speech System

Yao Qian, Frank Soong, Yining Chen, and Min Chu

Microsoft Research Asia, Beijing
{yaoqian, frankkps, ynchen, minchu}@microsoft.com

Abstract. In this paper we present our Hidden Markov Model (HMM)-based, Mandarin Chinese Text-to-Speech (TTS) system. Mandarin Chinese or Putonghua, "the common spoken language", is a tone language where each of the 400 plus base syllables can have up to 5 different lexical tone patterns. Their segmental and supra-segmental information is first modeled by 3 corresponding HMMs, including: (1) spectral envelop and gain; (2) voiced/unvoiced and fundamental frequency; and (3) segment duration. The corresponding HMMs are trained from a read speech database of 1,000 sentences recorded by a female speaker. Specifically, the spectral information is derived from short-time LPC spectral analysis. Among all LPC parameters, Line Spectrum Pair (LSP) has the closest relevance to the natural resonances or the "formants" of a speech sound and it is selected to parameterize the spectral information. Furthermore, the property of clustered LSPs around a spectral peak justify augmenting LSPs with their dynamic counterparts, both in time and frequency, in both HMM modeling and parameter trajectory synthesis. One hundred sentences synthesized by 4 LSP-based systems have been subjectively evaluated with an AB comparison test. The listening test results show that LSP and its dynamic counterpart, both in time and frequency, are preferred for the resultant higher synthesized speech quality.

Keywords: Speech synthesis, Trainable TTS, corpus-based TTS, statistics-based TTS, LSP.

1 Introduction

HMM-based speech synthesis has been successfully applied to TTS synthesis of many different languages, e.g. Japanese and English [1-3]. In this framework, the spectral envelop, fundamental frequency, and duration are modeled simultaneously by the corresponding HMMs. For a given text sequence, speech parameter trajectories and corresponding signals are then generated from the trained HMMs in the Maximum Likelihood (ML) sense. HMM is very effective to model the evolution of speech signals as a stochastic sequence of acoustic feature vectors. Many techniques have been developed for HMM-based speech recognition, e.g. context-dependent modeling, state-tying based on decision tree clustering, and speaker adaptation. They can be applied equally well to HMM-based speech synthesis in the sense of parameter trajectory generation.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 223 – 232, 2006.

The current performance of HMM-based speech synthesis has been further improved by using dynamic feature constraint in trajectory generation [3] and global variance for parameter generation [4], a high quality vocoder called STRAIGHT [5], and Hidden Semi-Markov Model duration model [6], , and trajectory model [16] or minimum generation error training [17]. Compared with the large corpus based concatenative speech synthesis, HMM-based speech synthesis is statistics based and vocoded. The speech generated from it is fairly smooth. Characteristics of the synthetic speech can be easily controlled by transforming HMM parameters in a statistically tractable metric like likelihood function. Furthermore, the small footprint of the HMM synthesizer has made it an ideal choice for an embedded system.

In this paper, we apply HMM-based speech synthesis to Mandarin, a syllabically paced tonal language. A tone-dependent phone set and corresponding phonetic and prosodic question set of decision tree are designed for HMM training. Line Spectrum Pair (LSP) [7], an alternative linear prediction parametric representation, is investigated as feature parameter to HMM-based speech instead of mel-ceptral features [8]. According to the properties of LSP [9], the speech generation module is revised correspondingly. The performances of four systems based on LSPs are tested in an AB comparison test. It shows that the S*ystem III*, which uses LSP and the dynamic features of adjacent LSP differences, achieves the better performance than the S*ystem II*, using the conventional method.

The rest of paper is organized as follows. In Section 2, HMM-based speech synthesis system is briefly illustrated; the representation and properties of LSP are introduced in Section 3; the speech parameter generation algorithm based on LSP is proposed in Section 4; Section 5 shows the experimental evaluation; and the conclusions are given in Section 6.

2 HMM-Based Speech Synthesis System

The schematic diagram of HMM-based Speech Synthesis system is shown in Figure 1 where both training and synthesis are shown.

In the training phase, the speech signal is converted to a sequence of observed feature vectors through the module of feature extraction and modeled by a corresponding sequence of HMMs. The observed feature vector consists of spectral parameters and excitation parameters, which are separated into different streams. The spectral feature comprises line spectrum pair (LSP) and log gain, and the excitation feature is log fundamental frequency. LSPs are modeled by continuous HMMs and F0s are modeled by multi-space probability distribution HMM (MSD-HMM) [10], which provides a cogent modeling of F0 without any heuristic assumptions or interpolations. Context-dependent phone models are used to capture the phonetic and prosody co-articulation phenomena. State typing based on decision-tree and minimum description length (MDL) [11] criterion is applied to overcome the problem of data sparseness in training. Stream-dependent models are built to cluster the spectral, prosodic and duration features into separated decision trees.

In the synthesis phase, input text is converted first into a sequence of contextual labels through the text analysis. The corresponding contextual HMMs are retrieved by traversing the trees of spectral and pitch information and the duration of each state is also obtained by traversing the duration tree, then the LSP, gain and F0 trajectories are generated by using the parameter generation algorithm based on maximum likelihood criterion with dynamic feature and global variance constraints. Finally, speech waveform is synthesized from the generated spectral and excitation parameters by LPC synthesis.

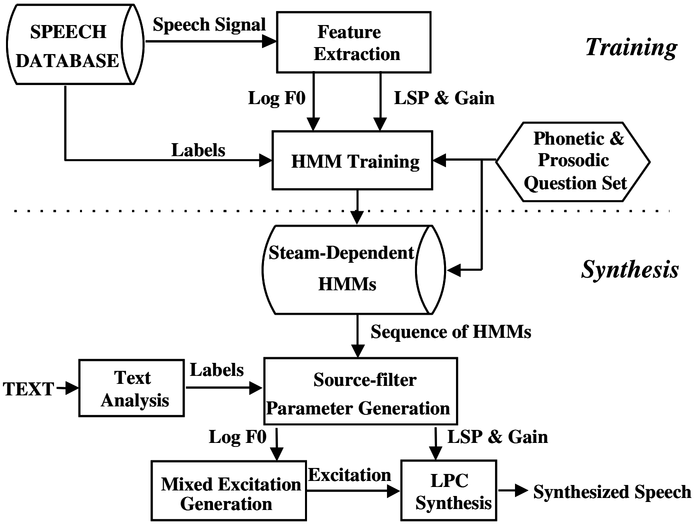


Fig. 1. HMM-based speech synthesis

3 The Properties of LSP

Line Spectrum Pair (LSP) [7] is an alternative linear prediction parametric representation. In LPC analysis, the speech signal is modeled as the output of an all-pole filter *H(z)* defined as

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - \sum_{i=1}^{M} a_i z^{-i}} \tag{1}$$

where M is the order of LPC analysis and $\{a_i\}_{i=1}^{M}$ are the corresponding LPC coefficients. The LPC coefficients can be represented by the LSP parameters, which are mathematically equivalent (one-to-one) and more amenable to quantization. LSP are calculated as follows:

$$P(z) = A(z) + z^{-(M+1)} A(z^{-1}) \tag{2}$$

$$Q(z) = A(z) - z^{-(M+1)} A(z^{-1}) \tag{3}$$

The symmetric polynomial *P(z)* and anti-symmetric polynomial *Q(z)* have the following two properties [9] : 1) All zeros of *P(z)* and *Q(z)* are on the unit circle; 2) zeros of *P(z)* and *Q(z)* are interlaced with each other. These properties are useful for finding the LSPs $\{\omega_i\}_{i=1}^{M}$, i.e., the roots the polynomial *P(z)* and *Q(z)*, which are ordered and bounded,

$$0 < \omega_1 < \omega_2 < \ldots < \omega_M < \pi \tag{4}$$

LSP has many advantages for speech representation [9,12,13]:

1) LSP parameters correlate well to "formant" or spectral peak location and bandwidth. The LPC power spectrum and the associated LSPs for vowel /a/ are shown in Figure 2, where clustered (two or three) LSPs depict a formant peak, in terms of both the center frequency and bandwidth.

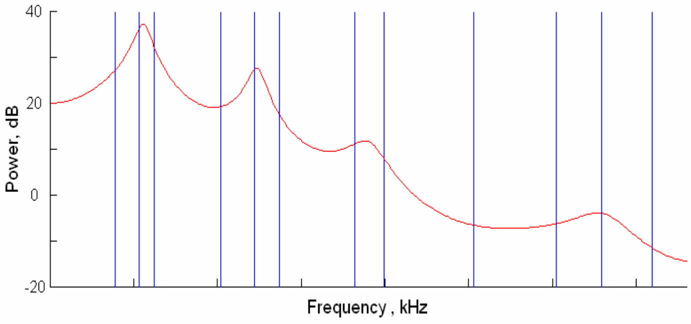


Fig. 2. LPC power spectrum and the associated LSPs for vowel /a/

2) Perturbation of an LSP parameter has a localized effect, i.e., a perturbation in a given LSP frequency only introduces a perturbation of LPC power spectrum in its neighborhood.
3) LSP parameter has a good interpolation property.

4 LSP Parameter Generation

In the HMM-based speech synthesis shown in Section 2, the speech parameter generation from given HMM state sequence is based on maximum likelihood criterion. In order to generate a smoother parameter trajectory, dynamic features are used as a constraint in the generation algorithm [3]. For a given HMM λ , it determines a speech parameter vector sequence $O = [C, \Delta C, \Delta^2 C]^T$, $C = [c_1^T, c_2^T, ..., c_T^T]^T$, $\Delta C = [\Delta c_1^T, \Delta c_2^T, ..., \Delta c_T^T]^T$, $\Delta^2 C = [\Delta^2 c_1^T, \Delta^2 c_2^T, ..., \Delta^2 c_T^T]^T$, which maximizes:

$$\begin{aligned} P(O \mid \lambda) &= \sum_{all\ Q} P(O,Q \mid \lambda) \\ &\simeq \max_{Q} P(O \mid Q,\lambda) P(Q \mid \lambda) \end{aligned} \tag{5}$$

If given state sequence $Q = \{q_1, q_2, q_3, ..., q_T\}$, Eq. 5 only need consider maximizing the logarithm of $P(O \mid Q,\lambda)$ with respect to $O = WC$, i.e.,

$$\frac{\partial LogP(WC \mid Q,\lambda)}{\partial C} = 0 \tag{6}$$

We obtain

$$W^T U^{-1} W C = W^T U^{-1} M \tag{7}$$

where

$$W=\begin{bmatrix} I_F \\ W_{\Delta F} \\ W_{\Delta\Delta F}\end{bmatrix}=\underbrace{\begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \cdots \\ \hdashline 0.5 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \cdots \\ \hdashline 1 & & & \\ & \ddots & & \\ & -2 & & \\ & & \ddots & \\ & & & \cdots \end{bmatrix}}_{DT}\begin{matrix}\left.\vphantom{\begin{matrix}1\\1\\1\\1\end{matrix}}\right\}DT\\ \left.\vphantom{\begin{matrix}1\\1\\1\\1\end{matrix}}\right\}DT\\ \left.\vphantom{\begin{matrix}1\\1\\1\\1\\1\end{matrix}}\right\}DT\end{matrix} \tag{8}$$

$$M = [m_{q_1}^T, m_{q_2}^T, ..., m_{q_T}^T]^T \tag{9}$$

$$U^{-1} = diag[U_{q_1}^{-1}, U_{q_2}^{-1}, ..., U_{q_T}^{-1}] \tag{10}$$

D is the dimension of feature vector and T is the total number of frame in the sentence. W is a block matrix which composes of three $DT \times DT$ matrices: Identity matrix (I_F), delta coefficient matrix ($W_{\Delta F}$) and delta-delta coefficient matrix ($W_{\Delta\Delta F}$). M and U are the $3DT \times 1$ mean vector and the $3DT \times 3DT$ covariance matrix, respectively.

As mentioned in Section 3, a gathering of (two or three) LSPs depicts a formant frequency and the closeness of the corresponding LSPs indicates the bandwidth of a given formant. Therefore, the distance between the adjacent LSPs is more critical than the absolute value of individual LSP. On the other hand, all LSP frequencies are ordered and bounded, i.e. any two adjacent LSP trajectories do not cross each other. Using static and dynamic LSPs in modeling and generation can not ensure the stability of LSPs. Consequently, we add the difference of adjacent LSP frequencies directly into spectral parameter modeling and generation. The *W*, which is used to transform the observation feature vector, is modified as

$$W = [I_F\ ,\ \ W_{DF}\ ,\ \ W_{\Delta F}\ ,\ \ W_{\Delta F}W_{DF}\ ,\ \ W_{\Delta\Delta F}\ ,\ \ W_{\Delta\Delta F}W_{DF}]^T \tag{11}$$

where F is static LSP; DF is the difference between adjacent LSP frequencies; ΔF and $\Delta\Delta F$ are dynamic LSPs, i.e., first and second order time derivatives; and W_{DF} is $(D-1)T \times DT$ matrix and constructed as

$$W_{DF} = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & & \cdots & \\ & & & & \ddots \\ & & & & \\ & & & & \end{bmatrix} \tag{12}$$

In this way, the correlation of adjacent LSPs can be modeled and diagonal covariance structure is still kept the same.

5 Experimental Evaluations

5.1 Experimental Setup

A broadcast news style speech corpus recorded by a female speaker is used in this study. The training data composes of 1,000 phonetically and prosodically rich sentences [14]; while the testing data consists of 100 sentences. Speech signal are sampled at 16 kHz, windowed by 25-ms window with a 5-ms shift, and transformed into 24th-order LSPs and their dynamic features in both frequency and time.

5-state,left-to-right HMMs with single, diagonal Gaussian distribution is adopted for phone model training. The phone set used is Ph97 [15], which achieved better performance than the other phone set in Mandarin tonal syllable recognition task. In Ph97, each Chinese tonal syllable is divided into a consonant followed by two consecutive tonal sonorant segments, e.g. /huang4/ is decomposed into /hu/, /aaH/ and /ngH/. Here, the glides like /i/ and /u/ are assigned to the Initial part and 2-scales (High/Low) pitch label is used instead of five numerical scales. The phone set designed in this way can carry tone information in the modeling at a little extra cost of the phone inventory size.

The phonetic and prosodic factors, which are used as question set in decision tree growing for contextual state tying, are listed as following

1) {preceding, current, succeeding} phone
2) Break index after the current word, three indices: minor, medium and major breaks, are used.
3) tone label (in 5-categories) of {preceding, current, succeeding} syllable
4) Position of a phone in a syllable
5) Position of a syllable in a "prosodic word" which are sandwiched by minor breaks
6) Position of a syllable in a breath group phrase which are limited by major breaks
7) Length of the current breath group phrase in terms of number of syllables

5.2 Experiments and Results

Four synthesis systems based on LSP features are built for comparison.

System I:
The W generating the observation feature in $O = WC$ is constructed as

$$W = [I_F\ ,\ \ W_{DF}\ ,\ \ W_{\Delta F}\ ,\ \ W_{\Delta F}W_{DF}\ ,\ \ W_{\Delta\Delta F}\ ,\ \ W_{\Delta\Delta F}W_{DF}]^T$$

where W is a $6DT \times DT$ matrix. Considering the higher orders of LSP are almost evenly spaced and most of speech formants are located below 4kHz, only the lower 16 out of 24 LSPs are used to compute DF (the distance between the adjacent LSPs). Therefore, the total dimension of observation feature vector is 126 (120 for LSP, 3 for gain and 3 for F0).

System II:
In this system, the observation feature vector is computed in conventional method. It consists of static, first and second order time derivatives. The corresponding W is defined as

$$W = [I_F\ ,\ \ W_{\Delta F}\ ,\ \ W_{\Delta\Delta F}]^T$$

The total dimension of observation feature vector is 78.

System III:
In order to make the results comparable with *system II* in feature dimensions, we only use the static and dynamic features of LSP difference (in frequency) as observation vector. The W is modified as

$$W = [I_F,\ \ W_{\Delta F}W_{DF},\ \ W_{\Delta\Delta F}W_{DF}]^T$$

Here, the total dimension of observation feature vector is equal to that of *system II*.

System IV:
40th -order LSP are used instead of 24th -order LSP in *System II*.

One hundred sentences are synthesized by the above four systems and evaluated in a subjective test. Fifty out of the one hundred sentences are randomly selected for an AB comparison preference test. Eight subjects are forced to choose one which sounds more natural from each pair. The results of the preference test are given in Figure 3, where shows:

a) *System I* achieves a better performance than *System II.* Modeling the difference of adjacent LSP frequency (*DF*) is very critical in reproducing the salient features of speech spectrum in HMM-based speech synthesis.
b) *System III* gives almost the same performance as *System I.* But the dimensionality of feature vector in *System III* is much less than that of *System I.* It indicates that with/without dynamic features of LSP frequency difference is critical to the performance of system.
c) S*ystem IV* slightly improve the performance comparing with *system II*, i.e., the performance improvement by using a higher order LSP is marginal.

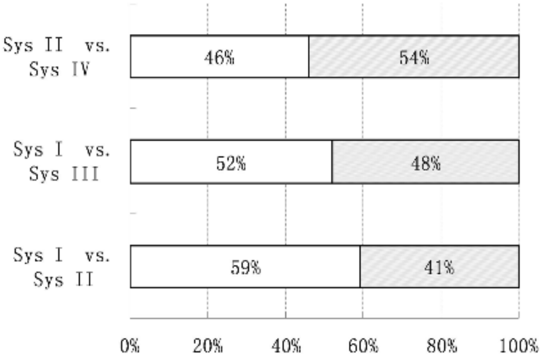


Fig. 3. The results of AB Test for four systems

5.3 Analysis

To analyze experimental results, we plot the spectra of synthesized and original speech signals for comparison. However, the duration of generated utterance can be different from that of the original since only means of state duration models are used in speech generation. An oracle experiment is designed to compare the spectra by isolating the effect of duration difference. A sequence of states, which are obtained by force-aligning the original feature observations with the spectral and pitch models, is used as Q in Eq. 6 for speech parameter generation. In this way, the spectra can be compared on a frame-by-frame basis between two different systems. An example of spectral comparison, LPC power spectra for vowel /u/ , is given in Fig. 4, where the bold dotted line, dotted line and solid line represent the spectra of the original, S*ystem II* and *System III*, respetively. The log power spectrum is plotted in dB scale and 25dB

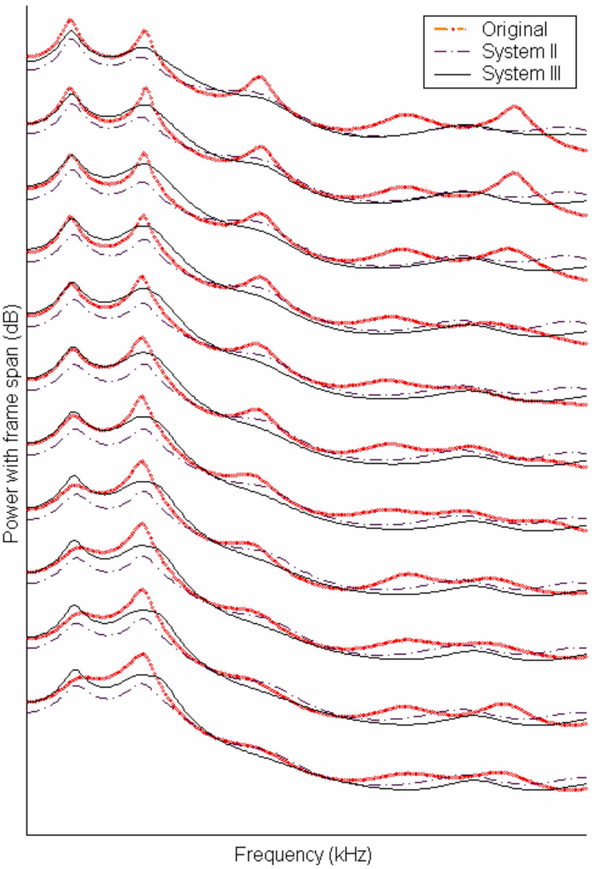


Fig. 4. LPC power spectra for vowel /u/ from original waveform, System II and System III

offset is used for separating adjacent frames. In Fig. 4 the formant structure of the generated spectra of S*ystem II* is sharper and closer to the original spectra than that of *System III*.

6 Conclusions

We present our HMM-based Text-to-Speech system for Mandarin Chinese synthesis in this paper. A tone-dependent phone set, Ph97, is employed in training HMMs with phonetic and prosodic question set in corresponding decision trees. We adopted LSP frequencies as acoustic spectral features for training HMMs. Subjective AB comparison preference test show that using LSPs and the dynamic features of adjacent LSPs in frequency considerably improve the quality of synthetic speech, in comparing with the conventional method.

References

[1] Zen, H. and Toda, T., An Overview of Nitech HMM-based Speech Synthesis System for Blizzard Challenge 2005, Proc. EuroSpeech, 2005.

[2] Tokuda, K., Zen, H. and Black, A.W., An HMM-based speech synthesis system applied to English,' 2002 IEEE Speech Synthesis Workshop, Santa Monica, California, Sep. 11-13, 2002.

[3] Tokuda, K., Kobayashi, T., Masuko, T., Kobayashi, T., and Kitamura, T., Speech Parameter generation algorithms for HMM-based speech synthesis. Proc. ICASSP, pp. 1315-1318, Istanbul, Turkey, June 2000.

[4] Tomoki, T. and Keiichi, T., Speech Parameter Generation Algorithm Considering Global Variance for HMM-Based Speech Synthesis, Proc. Eurospeech 2005.

[5] Kawahara, H., Masuda-Katsuse, I. and Cheveigne, A., Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based f0 extraction: possible role of a repetitive structure in sounds, Speech Communication, vol. 27, pp. 187–207, 1999.

[6] Zen, H., Tokuda, K., Masuko, T., Kobayashi, T. and Kitamura, T., Hidden semi-Markov model based speech synthesis, Proc. ICSLP, 2004, pp. 1185–1180.

[7] Itakura, F., Line spectrum representation of linear predictive coefficients of speech signals, J. Acoust. Soc. Am. 57 (Apr. 1975), S35.

[8] Fukada, T., Tokuda, K., Kobayashi, T, and Imai,S., An adaptive algorithm for mel-cepstral analysis of speech, Proc. ICASSP, 1992, pp. 137-140.

[9] Soong, F. K., and Juang, B. H. Line spectrum pair (LSP) and speech data compression. Proc. ICASSP, pp.1.10.1-1.10.4.,San Diego, CA, 1984.

[10] Tokuda, K., Masuko, T., Miyazaki, N., and Kobayashi, T., Multi-space Probability Distribution HMM, IEICE Trans. Inf. & Syst., E85-D(3):pp. 455-464, 2002.

[11] Shinoda, K. and Watanabe, T., Acoustic Modeling Based on The MDL Principle for Speech Recognition, Proc. EuroSpeech 1997, pp. 99-102.

[12] Wakita, H, Linear prediction voice synthesizers: line spectrum pairs (LSP) is the newest of the several techniques. Speech Technol. 1 (1981), pp.17-22

[13] Paliwal K. K. On the use of line spectral frequency parameters for speech recognition, Digital Signal Processing 2, pp 80-87 (1992)

[14] Chu, M., Peng, H., Yang, H. and Chang, E., Selecting non-uniform units from a very large corpus for concatenative speech synthesizer, Proc. ICASSP 2001,Salt Lake City.

[15] Huang, C., Shi, Y., Zhou, J. L., Chu, M., Wang, T., and Chang, E., Segmental Tonal Modeling for Phone Set Design in Mandarin LVCSR, Proc. ICASSP 2004, pp.901-904, 2004.

[16] Zen,H., Tokuda, K. and T. Kitamura, A Viterbi algorithm for a trajectory model derived from HMM with explicit relationship between static and dynamic features, Proc. of ICASSP 2004, pp. 837–840.

[17] Wu, Y.J. and Wang, R.H., Minimum generation error training for HMM-based speech synthesis. Proc. of ICAPP 2006, pp. 89-93

HMM-Based Emotional Speech Synthesis Using Average Emotion Model

Long Qin, Zhen-Hua Ling, Yi-Jian Wu, Bu-Fan Zhang, and Ren-Hua Wang

iFLYTEK Speech Lab, University of Science and Technology of China, Hefei
{qinlong, zhling, jasonwu, bfzhang}@mail.ustc.edu.cn
rhw@ustc.edu.cn

Abstract. This paper presents a technique for synthesizing emotional speech based on an emotion-independent model which is called "average emotion" model. The average emotion model is trained using a multi-emotion speech database. Applying a MLLR-based model adaptation method, we can transform the average emotion model to present the target emotion which is not included in the training data. A multi-emotion speech database including four emotions, "neutral", "happiness", "sadness", and "anger", is used in our experiment. The results of subjective tests show that the average emotion model can effectively synthesize neutral speech and can be adapted to the target emotion model using very limited training data.

Keywords: average emotion model, model adaptation, affective space.

1 Introduction

With the development of speech synthesis techniques, the intelligibility and naturalness of the synthetic speech has been improved a lot in the last decades. However, it is still a difficult problem for the TTS system to synthesize speech of various speakers and speaking styles with a limited database. It is known that the HMM-based speech synthesis can model speech for different speakers and speaking styles, and voice characteristics of the synthetic speech can be converted from one speaker to another by applying a model adaptation algorithm, such as the MLLR (Maximum Likelihood Linear Regression) algorithm, with a small amount of speech uttered by the target speaker [1], [2], [3]. Furthermore, the HMM-based emotional speech synthesis systems have been successfully constructed by directly training the models with enough emotion data or adapting the source model to the target emotion model when only a limited training data is available [4], [5].

We have realized a HMM-based speech synthesis system in which the LSP (Line Spectral Pair) coefficients and the STRAIGHT analysis-synthesis algorithm are employed [6], [7]. Then, by realizing the MLLR-based model adaptation algorithm, we provide our synthesis system with the ability of synthesizing voice of various speakers with different styles [8]. As only a very limited amount of emotion training data is acquired, we use the model adaptation method to construct our emotional speech system. Commonly, the source model for emotion adaptation is trained using only neutral speech data. But in this paper, we train an emotion-independent model using a

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 233 – 240, 2006.

multi-emotion speech database, which includes the neutral, happy and sad speech data of a female speaker. Compared with the neutral model, the average emotion model which considers the distributions of all emotions in the training data is a better coverage of the affective space. In fact, it takes the possible distribution of the target emotion into account, so it can achieve a better adaptation performance than the neutral model. The average emotion model is obtained using a shared decision tree clustering method which assures all nodes of the decision tree always have training data of all emotions [9]. Then we adapt the average emotion model to the target emotion model using a small amount of target speech data and generate the target synthetic speech.

In the following part of this paper, a description of our HMM-based emotional speech synthesis system is presented in section 2. Section 3 presents the speech database information, the training set design and the results of subjective experiments, while section 4 provides a final conclusion.

2 System Description

The framework of our HMM-based emotional speech synthesis system, shown in Figure 1, is the same as the conventional HMM-based synthesis system except that an average emotion model is used as the source model and a MLLR-based model adaptation stage, using context clustering decision tree and appropriate regression matrix, is added between the training stage and the synthesis stage.

In the training stage, the LSP coefficients and the logarithm of fundamental frequency are extracted by the STRAIGHT analysis. Afterwards, their dynamic features including delta and delta-delta coefficients are calculated. The MSD (multi-space probability distribution) HMMs are introduced to model spectrum and pitch patterns because of the discontinuity of pitch observations [10]. And state durations are modeled by the multi-dimensional Gaussian distributions [11]. To obtain the average emotion model, firstly, the context-dependent models without context clustering are separately trained for each emotion. Then all these context-dependent emotion models are clustered using a shared decision tree and the Gaussian pdfs of the average emotion model is calculated by tying all emotions' Gaussian pdfs at every node of the tree. Finally, state duration distributions of the average emotion model are obtained under the same clustering procedure.

In the adaptation stage, the spectrum, pitch and duration HMMs of the average emotion model are all adapted to those of the target emotion. To achieve supersegmental feature adaptation, the context decision tree constructed in the training stage is used to tie regression matrices. And because of the correlations between the LSP coefficients of adjacent orders, the appropriate regression matrix format is adopted according to the different amount of training data. At first, the spectrum and pitch HMMs are adapted to the target emotion HMMs. Then, on the basis of the converted spectrum and pitch HMMs, the target emotional utterances are segmented to get the duration adaptation data. So that the duration model adaptation can be achieved.

In the synthesis stage, according to the given text to be synthesized, a sentence HMM is constructed by concatenating the converted phoneme HMMs. From the sentence HMM, the LSP and pitch parameter sequences are obtained using the speech parameter generation algorithm, where phoneme durations are determined based on the state duration distributions. Finally, the generated parameter sequences of spectrum, converted from the LSP coefficients, and F0 are put into the STRAIGHT decoder to synthesize the target emotion speech.

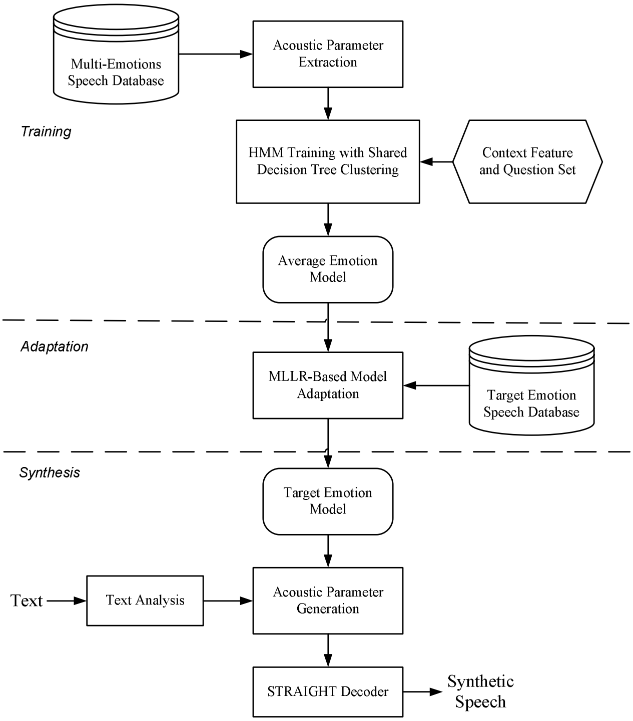

Multi-Emotions Speech Database
Acoustic Parameter Extraction
Training
HMM Training with Shared Decision Tree Clustering
Context Feature and Question Set
Average Emotion Model
Adaptation
MLLR-Based Model Adaptation
Target Emotion Speech Database
Synthesis
Target Emotion Model
Text
Text Analysis
Acoustic Parameter Generation
STRAIGHT Decoder
Synthetic Speech

Fig. 1. HMM-based emotional speech synthesis system

3 Experiment and Evaluation

3.1 Speech Database

We constructed a multi-emotion Chinese speech database of a female speaker including four emotions, “neutral”, “happiness”, “sadness” and “anger”. There are phonetically balanced 1200 sentences for “neutral” and 400 sentences for each of the other emotions. Contexts of all the emotion samples are different from each other. Firstly, we evaluated whether the recorded speech samples were uttered in the intended emotions. All the speech samples were randomly presented to ten listeners, and the listeners were asked to select an emotion from the four emotions. The listeners were asked to recognize the emotion of speech samples not by contexts but by acoustic presentations. Table 1 shows the classification rates for each emotion of the recorded speech. We can find that most of the recorded speech can successfully represent the intended emotions.

Table 1. Classification results of the recorded natural speech

	Classification (%)			
	Neutral	Happy	Sad	Angry
Neutral	96.0	2.0	1.0	1.0
Happy	7.0	85.5	0.5	7.0
Sad	5.0	0	91.0	4.0
Angry	1.5	6.0	1.0	91.5

3.2 Training Set Design

In order to realize an average emotion model, a good coverage for the affective space of the training data is expected. The affective space can be described with Russell's circumplex model [12], [13]. As illustrated in Figure 2, Russell has developed a two dimensional circumplex model of affection that makes it straightforward to classify an emotion as close or distant from another one. He called the two dimensions "valence" and "arousal". These terms correspond to a positive/negative dimension and an activity dimension respectively. As the multi-emotion database can only contain several kinds of emotions sampled from the affective space, it is important to choose the most representative emotions for training. In our experiment, the multi-emotion database has four emotions, neutral, happiness, sadness, and anger. We decide to use the speech data of neutral, happiness and sadness as the training data for the average emotion model, because happiness that is a very positive emotion with high arousal and sadness that is a very negative emotion with low arousal almost are two corresponding emotions and can be a rational representation of the affective space. Meanwhile, the angry speech data is left for model adaptation and evaluation.

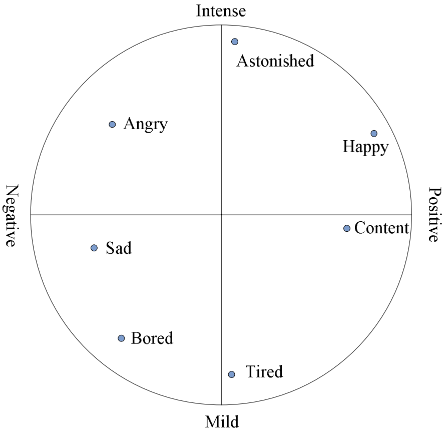


Fig. 2. Circumplex model of affect as described by Russell (1980)

3.3 Experimental Conditions

The average emotion model is trained by 300 sentences of each emotion, including neutral, happy and sad, selected from the multi-emotion database. A neutral model is trained by 1000 neutral sentences selected from the multi-emotion database for comparison. And 100 angry sentences are used for the model adaptation and evaluation. The speech is sampled at a rate of 16KHz. Spectrum and pitch is obtained by the STRAIGHT analysis. Then they are converted to the LSP coefficients and the logarithm F0 respectively, and their dynamic parameters are calculated. Finally, the feature vector of spectrum and pitch is composed of the 25-order LSP coefficients including the zeroth coefficient, the logarithm F0, as well as their delta and delta-delta coefficients. We use the 5-state left-to-right no-skip HMMs in which the spectral part of each state is modeled by the single diagonal Gaussian output distributions. The duration feature vector is a 5 dimensional vector, corresponding to the 5-state HMMs, and the state durations are modeled by the multi-dimensional Gaussian distributions.

3.4 Experiments on the Average Emotion Model and the Neutral Model

Table 2 shows the number of distributions of the average emotion model and the neutral model after decision tree context clustering. Here, we set the weight for adjusting the number of parameters of the model during the shared decision tree context clustering as 0.6. From the table, it can be seen that the two models have comparable distributions.

Table 2. The number of distributions after context clustering

	Neutral Model	Average Emotion Model
Spectrum	3247	3115
F0	4541	5020
Duration	599	589

50 sentences of the synthetic speech generated by each model were also presented to 10 listeners to choose the emotion from the four emotions and the result is illustrated in Table 3. It can be found that both the two models can effectively synthesize neutral speech. However, the result of the neutral model is a little better than that of

Table 3. Classification results of the synthetic speech generated by the neutral model and the average emotion model

	Classification (%)			
	Neutral	Happy	Sad	Angry
Neutral Model	92.2	5.7	2.1	0
Average Emotion Model	84.2	5.0	10.1	0.7

the average emotion model. Some of the synthetic speech generated by the average emotion was misrecognized as sad. That may be because sadness has a better expression than happiness in the training data, as shown in Table 1, so that the average emotion model has a slight bias towards sadness.

3.5 Experiments on the Emotion Adaptation

In the model adaptation stage, the neutral model or the average emotion model is adapted to the target emotion model with 50 angry sentences which are not included in the adaptation training data. The 3-block regression matrix is adopted and the regression matrices are grouped using a context decision tree clustering method. First, 10 listeners were asked to recognize the emotion of 50 synthetic speech samples generated by the two methods from the four emotions. The classification results are presented in Table 4. It can be found that about 70% of the synthetic speech can by successfully recognized by the listeners and the average emotion model has a better adaptation performance.

Table 4. Classification results of the synthetic speech generated by the angry model adapted from the neutral model and the average emotion model

	Classification (%)			
	Neutral	Happy	Sad	Angry
Neutral Model	16.7	2.3	10.4	70.6
Average Emotion Model	13.1	3.4	10.0	73.5

Compared to the speech synthesized by the adapted average emotion model, some speech samples generated by the adapted neutral model sound to be not natural especially in prosody. Figure 3 demonstrates the F0 contours of the synthetic speech generated from the adapted neutral model and the adapted average emotion model respectively, meanwhile the F0 contour of the target speech is also presented. The dotted red line presents the F0 contour generated from the adapted neutral model while the solid

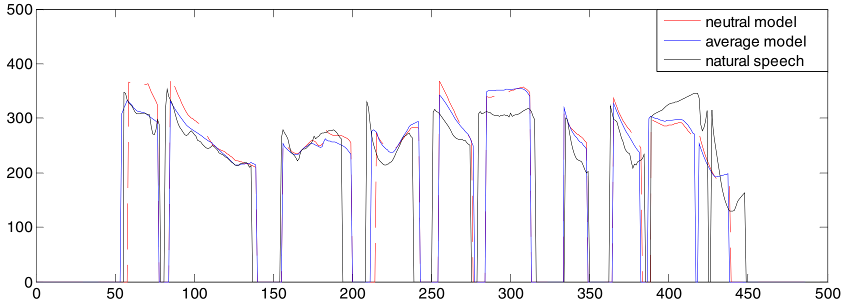


Fig. 3. Comparison of F0 contours generated by the angry model adapted from the neutral model and the average emotion model

blue line is the result of the adapted average emotion model and the solid black line is the F0 contour of target speech. We can see that the values of F0 generated from the adapted average emotion model are more similar to those of the target speech.

4 Conclusion

A HMM-based emotional speech synthesis system is realized using a model adaptation method. At first, an average emotion model is trained using a multi-emotion speech database. Then, the average emotion model is adapted to the target emotion model with a small amount of training data using a MLLR-based model adaptation technique in which a context decision tree is built to group HMMs of the average emotion model. To compare the performance of the proposed method, a neutral model is also trained and adapted. From the results of the subjective tests, it can be seen that both methods can effectively synthesize the intended emotion speech. In addition, the adaptation performance of the average emotion model is slightly better than that of the neutral model.

If having more emotional speech data, there will be a better coverage of the affective space, so we can train a more reasonable average emotion model. Our future work will focus on increasing the number of emotion categories in the multi-emotion database and improving the performance of the average emotion model. At the same time, various emotions will be selected as the target emotion to evaluate the effectiveness of the average emotion model.

Acknowledgement

This work was partially supported by the National Natural Science Foundation of China under grant number 60475015.

References

1. T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, “Speech synthesis from HMMs using dynamic features,” Proc. ICASSP-1996, pp. 389-392, 1996.
2. C.J. Leggetter and P.C. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models,” Computer Speech and Language, vol.9, no.2, pp. 171-185, 1995.
3. T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, “Speaker adaptation for HMM-based speech synthesis system using MLLR,” The Third ESCA/COCOSDA Workshop on Speech Synthesis, pp. 273-276, Nov. 1998.
4. J. Yamagishi, K. Onishi, T. Masuko, and T. Kobayashi, “Acoustic modeling of speaking styles and emotional expressions in HMM-based speech synthesis,” IEICE Trans. Information and Systems, vol. E88-D, no.3, pp.502-509, March 2005.
5. J. Yamagishi, M. Tachibana, T. Masuko, and T. Kobayashi, “Speaking style adaptation using context clustering decision tree for HMM-based speech synthesis,” Proc. ICASSP-2004, vol.1, pp. 5-8, May 2004.

6. H. Kawahara, “Restructuring speech representations using a pitch-adaptive time frequency smoothing and a instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sound”, Speech Communication 27, pp. 187-207, 1999.
7. Y.J. Wu and R.H. Wang, “HMM-based trainable speech synthesis for Chinese,” to appear in Journal of Chinese Information Processing.
8. Long Qin, Yi-Jian Wu, Zhen-Hua Ling, and Ren-Hua Wang, “Improving the performance of HMM-base voice conversion using context clustering decision tree and appropriate regression matrix,” to appear in Proc. ICSLP-2006.
9. J. Yamagishi, M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, “A context clustering technique for average voice models,” IEICE Trans. Information and Systems, vol. E86-D, no. 3, pp. 534-542, March 2003.
10. K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, “Hidden Markov models based on multi-space probability distribution for pitch pattern modeling,” Proc. ICASSP-1999, pp. 229-232, Mar. 1999.
11. T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Duration modeling for HMM-based speech synthesis,” Proc. ICSLP-1998, vol.2, pp. 29-32, Nov. 1998.
12. J.A. Russell, “A circumplex model of affect,” Journal of Personality and Social Psychology, vol. 39, pp. 1161-1178, 1980.
13. R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J.G. Taylor, “ Emotion recognition in human-computer interaction,” IEEE Signal Processing Magazine, Vol. 18, Issue 1, pp. 32-80, Jan. 2001.

A Hakka Text-To-Speech System

Hsiu-Min Yu[1], Hsin-Te Hwang[2], Dong-Yi Lin[2], and Sin-Horng Chen[2]

[1] Department of Foreign Languages, Chung Hua University, Hsinchu
[2] Department of Communication Engineering, National Chiao Tung University, Hsinchu
schen@cc.nctu.edu.tw

Abstract. In this paper, the implementation of a Hakka text-to-speech (TTS) system is presented. The system is designed based on the same principle of developing a Mandarin and a Min-Nan TTS systems proposed previously. It takes 671 base-syllables as basic synthesis units and uses a recurrent neural network (RNN)-based prosody generator to generate proper prosodic parameters for synthesizing natural output speech. The whole system is implemented by software and runs in real-time on PC. Informal subjective listening test confirmed that the system performed well. All synthetic speeches sounded well for well-tokenized texts and fair for texts with automatic tokenization.

1 Introduction

In Taiwan, there are three major languages including Mandarin, Min-Nan/Taiwanese and Hakka. Over 90% people speak Mandarin, about 70% whose mother-tongue are Taiwanese, and only 10% speak Hakka. In the past, there are only few studies in both Min-Nan and Hakka TTS [1-5]. This is due in part to the lack of a unified written-form representation and in part to the difficulty of collecting a large speech database. In recent years, Taiwan government started to pay more attention to mother-tongue education such as Taiwanese and Hakka. This motivates us to accomplish the Min-Nan and Hakka TTS system. In previous studies, we have built a Mandarin and a Min-Nan TTS systems [1-6]. In this study, we try to use the same technique to implement a Hakka TTS system.

Hakka includes several sub-dialects, namely Hoi-Liuk, Si-Rhan, Ta-Pu, etc. We choose Si-Rhan in this study. Just like Mandarin and Min-Nan, Hakka speech is a monosyllabic and tonal language. Each character is pronounced as a syllable carrying a lexical tone. There are only 671 base-syllables and 6 tones. Fig. 1 shows typical pitch frequency contours of these 6 tones. These 671 base-syllables also have the same *initial-final* structure like Mandarin and Min-Nan base-syllables. There are 17 *initials* and 72 *finals*. Although Hakka speech has similar linguistic characteristics as Mandarin speech, it does not have a standard written form like Min-Nan speech. The written form is a hybrid one which uses Chinese characters to represent ordinary words and represents some extraordinary syllables in the Romanization form. Unfortunately, the system to represent words in Chinese characters is still not

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 241 – 247, 2006.

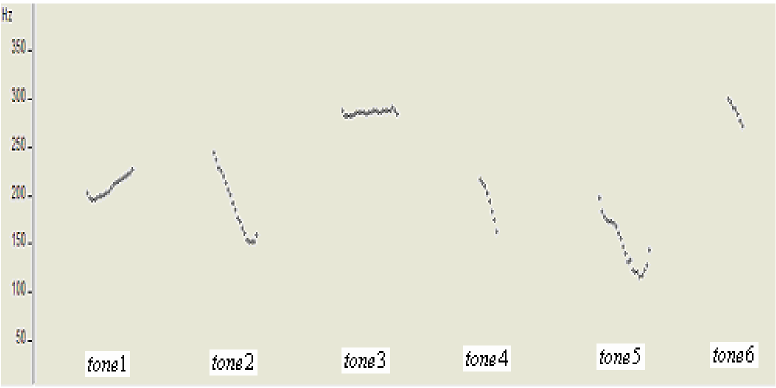


Fig. 1. Typical pitch contours of 6 tones for Si-Rhan Hakka

standardized nowadays in Taiwan. This makes the text analysis very difficult for Hakka language.

In this paper, a study of developing a Hakka TTS system is presented. Since Hakka speech has similar linguistic characteristics as Mandarin and Min-Man speech, the system is designed based on the same principle as that used in our previous Mandarin and Min-Nan TTS system developments [1-6]. The system consists of four main parts: text analyzer, RNN-based prosody generator, waveform table, and PSOLA speech synthesizer. It tokenizes the input text into a word sequence in text analyzer, takes all 671 base-syllables as the basic synthesis units and stored in waveform table, adopts the RNN-based approach to generate prosodic parameters, and uses the PSOLA synthesis method to generate the output synthetic speech.

The paper is organized as follows. Section 2 presents the proposed Hakka TTS system. All functional blocks of the system are discussed in detail. Experimental results to evaluate the system performance is discusses in Section 3. Some conclusions are given in the last section.

2 System Description

Fig. 2 shows a block diagram of the proposed Hakka TTS system. The system consists of four main parts: a text analyzer, an RNN-based prosodic generator, a waveform table of 671 base-syllables, and a PSOLA speech synthesizer. Input text is first analyzed in the text analyzer to obtain word, POS and syllable sequences. The basic waveform sequence corresponding to the syllable sequence is then formed by looking up the waveform table. Some linguistic features are extracted from the word and POS sequences and used in the RNN-based prosody generator to generate prosodic parameters. Lastly, the basic waveform sequence is modified in PSOLA by using the prosodic parameters to generate the output synthetic speech. We discuss these four main blocks in detail as follows.

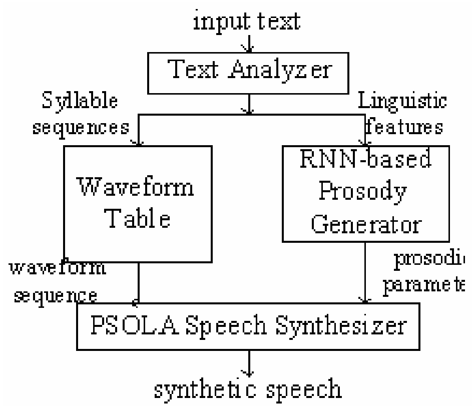


Fig. 2. A block diagram of the Hakka TTS system

2.1 Text Analyzer

The task of the text analyzer is to analyze the input text to extract some linguistic features. In this study, input text is represented in a hybrid written form of Hakka. Each text is a concatenation of words. Each word is a concatenation of Chinese characters and/or monosyllables represented in Romanization form. Each Romanized monosyllable is an alphabet sequence. The text analyzer first tags the input text to obtain the word sequence represented by the Big-5 code by using a statistical model based method to find the best word and POS sequence simultaneously. Here, a 34541-word lexicon, containing 1- to 8-syllabic words, and a long-word-first criterion are employed to segment and tokenize the Big-5 code sequence into word sequence. POS bigram model calculated from a database containing utterances of short sentences and paragraphic text are used in the tagging process.

After obtaining the optimal word and the associated POS sequence, we then use two additional bracketing rules to construct two types of compound words which are not contained in the lexicon. One is for the character-duplicated compound word and the other is for determiner-measure compound word.

Two sets of linguistic features are then extracted from the word sequence. One is the syllable sequence, which is extracted from the word sequence by looking up the lexicon. It will be used in waveform table to obtain the basic waveform sequence. The other consists of two subsets of syllable-level and word-level linguistic features and is used in the RNN-based prosody generator to synthesize proper prosodic parameters. The subset of syllable-level linguistic features contains four sequences of the initial consonant types of the syllables, the final vowel types of the syllables, tones of the syllables, and the position of the syllables in the corresponding words. The subset of word-level linguistic features includes the POS sequence, word lengths and punctuation marks.

2.2 RNN-Based Prosody Generator

The function of the RNN-based prosody generator is to produce proper prosodic parameters by using the linguistic features generated by the text analyzer. Fig. 3 shows a block diagram of the RNN-based prosody generator. The RNN has the same architecture as the one used in our previous Mandarin TTS and Min-Nan TTS studies. It is a four-layer network with one input layer, two hidden layers, and one output layer. It generates all prosodic parameters required in our system. They include pitch contour of syllable, energy level of syllable, initial and final durations of syllable, and inter-syllable pause duration. It can be functionally decomposed into two parts. The first part consists of the input layer and the first hidden layer and is taken as a prosodic model to explore the prosodic phrase structure of the synthetic speech by using the input word-level linguistic features. It operates in a word-synchronous mode using word-level input linguistic features including 47 types of POS, word lengths, and 2 types of punctuation mark extracted from the context of the current word. The second part consists of the second hidden layer and the output layer. It operates in a syllable-synchronous mode using syllable-level input linguistic features including tones, *initial* types, *final* types, and syllable location in a word extracted from the context of the current syllable. In this study, RNN of this architecture has been proven in previous studies to be effective on exploring the contextual information of the input linguistic features for generating proper output prosodic parameters. So we choose it in this study.

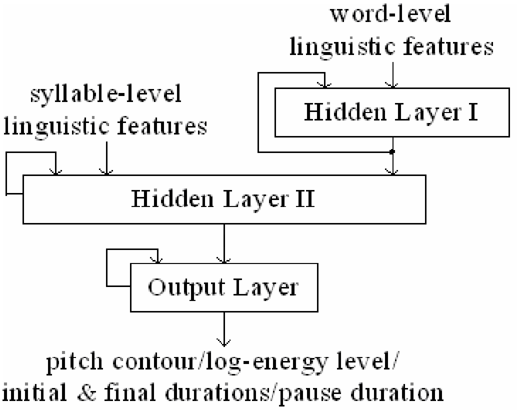


Fig. 3. A block diagram of the RNN-based prosody generator

2.3 Waveform Table

The function of the waveform table is to provide the basic primitive waveforms of the synthetic speech. It stores waveform templates of all 671 base-syllables, which are the basic synthesis units used in our system. All these waveform templates are obtained from isolated-syllable utterances pronounced clearly by a female speaker. All speech signals are direct digital recorded using a PC with a sound card. The sampling rate is 16 kHz. In synthesis, all constituent waveform templates of the input syllable sequence are extracted from the waveform table, directly concatenated together, and sent to PSOLA for prosody modification.

2.4 The PSOLA Speech Synthesizer

The PSOLA speech synthesizer is widely used in TTS. It can generate high quality synthetic speech in low computational complexity. The function of the PSOLA speech synthesizer is to generate the output synthetic speech by modifying the input basic primitive waveform sequence to make its prosodic parameters match the target ones generated by the RNN prosody generator. Prosody modifications include changing the pitch contour for each syllable, adjusting the durations of the initial consonant and the final vowel of each syllable, scaling the energy level of each syllable, and setting the inter-syllable pause duration. Finally, output the synthetic speech from a 16-bit Sound Blaster card.

3 Experimental Results

Performance of the proposed Hakka TTS system was examined by simulation using a single female speaker database. The database contains 316 utterances. The total number of syllables is 47408. Besides, a set of 671 isolated base-syllable utterances was recorded for developing the waveform table. All speech signals were digitally recorded in a 16 kHz rate. All the speech signals and the associated texts were manually pre-processed in order to extract the acoustic features and the linguistic features required to train and test the system.

We first examined the performance of the RNN prosody generator. Table 1 lists the root mean square errors (RMSEs) of the synthesized prosodic parameters. Comparing with those obtained in [6] for Mandarin TTS, these RMSEs are a little worse. This may come from the larger variations of the prosodic features in this Hakka speech database. Fig. 4 shows a typical example of the synthesized prosodic parameters. It can be seen from the figure that the synthesized prosodic parameters of most syllables matched well with their original counterparts. To evaluate the performance of Text Analyzer, Table 2 shows the performance of word segmentation.

The whole system was implemented by software on a PC with a 16-bit Sound Blaster card. An informal subjective listening test using various texts which were not included in the database was finally derived to examine the performance of the

Table 1. The RMSEs of the synthesized prosodic parameters

	Inside Test	Outside Test
F0 Contour	1.9ms	2.2ms
Pause Duration	56.8ms	65.4ms
Initial Duration	20.7ms	25.6ms
Final Duration	42.9ms	45.7ms
Energy Level	3.7dB	4.3dB

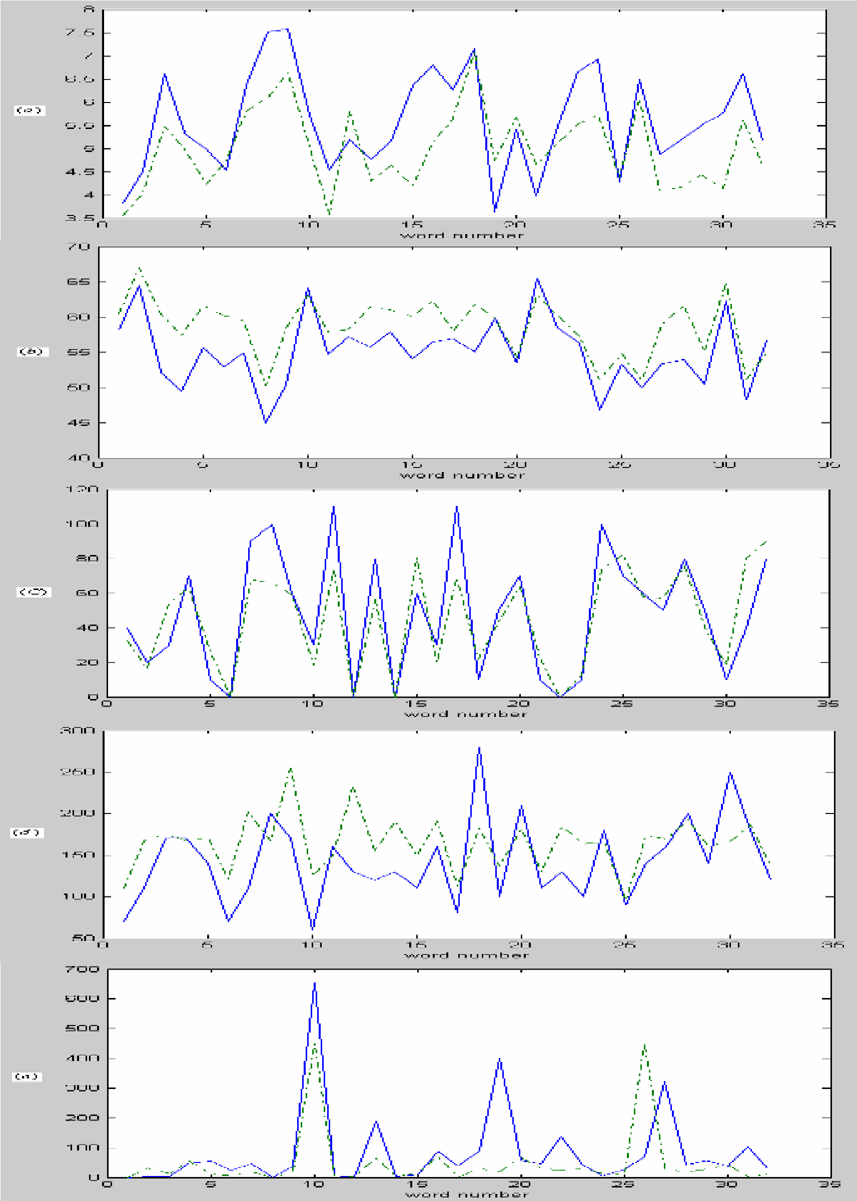


Fig. 4. A typical example: the synthesized sequences of (a) pitch mean(ms), (b) energy level(dB), (c) initial duration(ms), and (d) final duration(ms) of syllables as well as (e) inter-syllable pause duration(ms). The text is “記得我看過一篇文章，它內容大約是講客家是中國按多民族中最進步介民族”.

system. Many native Chinese living in Taiwan whose mother-tongue is Hakka confirmed that all synthesized speeches sounded well for well-tokenized texts and fair for texts with automatic tokenization.

Table 2. Performance for word segmentation

N1(hand-segmented word number)	25306
N2(TA-segmented word number)	26039
N3(correct TA-segmented word number)	20910
Recall(N3/N1)	82.63%
Precision(N3/N2)	80.3%

4 Conclusions

We have presented the implementation of a Hakka TTS system in this paper. The system was designed based on the same principle of developing a Mandarin and a Min-Nan/Taiwanese TTS systems proposed previously. Experimental results confirmed that the system performed well. Further studies to improve the naturalness of the synthetic speech by incorporating a more sophisticated text analysis scheme and by adding some tone sandhi rules are worthwhile doing in the future.

Acknowledgement

This work was supported by Council for Hakka Affairs Executive Yuan, Republic of China (Taiwan), and also partially supported by NSC, ROC. The authors thank H. M. Yu and W. Z. Gong for supplying the lexicon and the text corpus.

References

1. H. Fu, "Automatic Generation of Synthesis Units for Taiwanese Text-to-Speech System," Master Thesis, EE Dept., Chang Gung University, June 2000.
2. Y. J. Sher, K. C. Chung and C. H. Wu, "Establish Taiwanese 7-tones Syllable-based Synthesis units Database for the Prototype Development of Text-to-Speech System," in Proceedings of ROCLING XII, August 1999.
3. Y. C. Yang, "An Implementation of Taiwanese Text-to-Speech System," Master Thesis, Communication Engg. Dept., National Chiao Tung Univ., June 1999.
4. S. H. Chen and C. C. Ho, "A Min-Nan Text-to-Speech System," ISCSLP'2000, Beijing, Oct. 2000.
5. Wei-Chih Kuo, Xiang-Rui Zhong, Yih-Ru Wang and Sin-Horng Chen, "A High-Performance Min-Nan/ Taiwanese TTS System", ICASSP2003.
6. S. H. Chen, S. H. Hwang and Y. R. Wang, "An RNN-Based Prosodic Information Synthesizer for Mandarin Text-to-Speech," IEEE Trans. Speech and Audio Processing, vol. 6, no. 3, pp.226-239, 1998.
7. S. Haykin, *Neural networks – A comprehensive foundation*, Macmillan College Publishing Company, 1994.

Adaptive Null-Forming Algorithm with Auditory Sub-bands*

Heng Zhang, Qiang Fu, and Yonghong Yan

ThinkIT Speech Lab.,
Institute of Acoustics, Chinese Academy of Sciences
{hzhang, qfu, yyan}@hccl.ioa.ac.cn

Abstract. This paper presents a modified noise reduction algorithm for speech enhancement based on the scheme of null-forming. A fixed infinite-duration impulse response (IIR) filter is designed to calibrate the mismatch of the microphone pair. To weaken the performance degradation caused by narrow-band effect, the signal is decomposed into several specified sub-bands with auditory characters. This increases the signal to noise ratio (SNR) considerably while preserving the auditory effect. Experiments are carried out to show the effectiveness of these processes.

1 Introduction

The performance of speech communication and automatic speech recognition (ASR) system is often disturbed by environmental noise. Many techniques featuring microphone arrays have been used to improve the performances mentioned above by enhancing desired speech signal while suppressing noise and interference. Some of these techniques are also of great help to hearing aids.

With the help of microphone arrays, we can choose to focus on signals from a particular direction [1]. Better estimation of signal and noise can also be achieved. The Frost beamformer [2] was one of the first array structures to handle adaptive broadband processing by canceling everything that does not come from the look direction. Later, Griffiths and Jim developed an alternative method [3] called generalized sidelobe canceler (GSC) which effectively reduces the computational complexity as well as provides flexibility to implement beamformers according to different designing principles by using the GSC-structure [4]. Other algorithms include Zelinski's approach of post-filtering [5], which employs auto- and cross- correlation functions of signals to estimate the power spectral density (PSD) of signals and noise.

The scheme of adaptive null-forming based on differential microphone technique was first put forward by Elko and Pong in 1995 [6], and developed by Luo et al. [7], which features a simple structure employing two omni-directional microphones in end-fire orientation. Compared to beamforming algorithms using

* This work is (partly) supported by Chinese 973 program (2004CB318106), National Natural Science Foundation of China (10574140, 60535030), and Beijing Municipal Science & Technology Commission(Z0005189040391).

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 248–257, 2006.

more than 4 microphones, it is hard to achieve sharp receiving pattern with less sensors. Therefore, instead of trying to form a narrow beam aiming at the speech source, null-forming focuses on forming a receiving pattern with a null steered to the noise source adaptively while maintaining the desired signal coming from the front. This algorithm is very effective and can handle non-stationery noise, but is sensitive to the mismatch of the microphone pair. On encountering great reverberation or more than one noise sources, the performance of the system will drop to a certain extent.

In this paper, we present a modified algorithm based on adaptive null-forming with auditory sub-bands. A fixed IIR filter is used to calibrate the mismatch of the microphone pair. To weaken the performance degradation caused by narrow-band effect, we decompose the signal into several sub-bands according to auditory masking effect. This increases SNR considerably while preserving the auditory effect.

2 Single-Band Adaptive Null-Forming Scheme

The adaptive null-forming algorithm, with two microphone in end-fire orientation, is shown in Fig. 1, in which

$Fore$	signal received by microphone in the front
$Back$	signal received by microphone in the back
θ	signal arrival angle
c	propagation speed of sound wave
$x(n)$	first-order differential result of upper branch
$y(n)$	first-order differential result of lower branch
$W(n)$	coefficient of adaptive filter
$z(n)$	system output

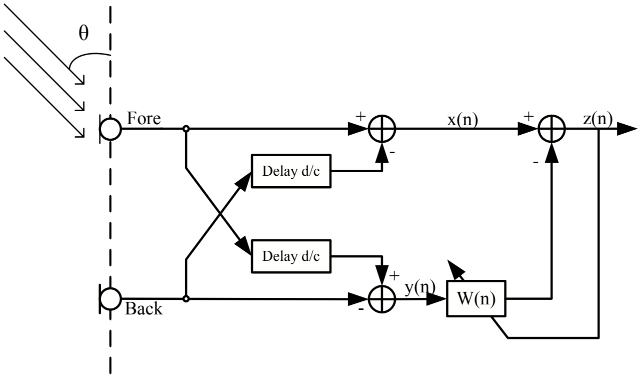


Fig. 1. Adaptive null-forming

We take the front microphone as a reference and have $x(n)$,$y(n)$ and $z(n)$ as

$$x(n) = 1 - exp\left(-j2\pi f\frac{d}{c}(1+\cos\theta)\right) \tag{1}$$

$$y(n) = exp\left(-j2\pi f\frac{d}{c}\right) - exp\left(-j2\pi\frac{d}{c}\cos\theta\right) \tag{2}$$

$$\begin{aligned} z(n) = 1 - exp\left(-j2\pi f\frac{d}{c}(1+\cos\theta)\right) \\ - W(n) \times \left(exp\left(-j2\pi f\frac{d}{c}\right) - exp\left(-j2\pi\frac{d}{c}\cos\theta\right)\right) \end{aligned} \tag{3}$$

where d is the spacing between the two microphones. All the right terms of the equations above should be multiplied by the signal received by the front microphone. The power of system output $z(n)$ can be calculated.

$$R_z(n) = z(n)z^*(n) \tag{4}$$

Insert (3) into (4), then R_z is 0 at a certain degree of arrival, θ_{null}, if

$$\sin\left(\pi f\frac{d}{c}(1+\cos\theta_{null})\right) + W(n)\left(\sin\left(\pi f\frac{d}{c}(1-\cos\theta_{null})\right)\right) = 0 \tag{5}$$

The equation above can be rearranged to yield

$$W(n) = -\frac{\sin\left(\pi f\frac{d}{c}(1+\cos\theta_{null})\right)}{\sin\left(\pi f\frac{d}{c}(1-\cos\theta_{null})\right)} \tag{6}$$

With the approximation $\sin\theta \approx \theta$ within the frequency range of interest, (6) can be approximated as

$$W(n) = -\frac{1+\cos\theta_{null}}{1-\cos\theta_{null}} \tag{7}$$

Only the interval $[0°, 180°]$ will be taken into consideration because of the periodicity of $\cos\theta$.

$$\frac{dW(n)}{d\theta_{null}} = 2\frac{\sin\theta_{null}}{(1-\cos\theta_{null})^2} \geq 0 \qquad \text{for} \quad 0° < \theta_{null} \leq 180° \tag{8}$$

Equation (8) shows that the relation between the null and $W(n)$ is monotonic. That is to say, a unique angle of null can be obtained given a $W(n)$, which can be calculated adaptively as demonstrated below.

$$\begin{aligned} E[z^2(n)] &= E[(x(n) - W(n)y(n))^2] \\ &= R_{xx} - 2W(n)R_{xy} + W^2(n)R_{yy} \end{aligned} \tag{9}$$

Minimizing (9) leads to

$$W_{opt} = \frac{R_{xy}}{R_{yy}} \tag{10}$$

which can be calculated iteratively [7].

3 Calibration of the Microphones

The conclusion of Sect.2 is under the assumption that the two microphones are strictly identical, which can be hardly satisfied in practice. Mismatch of the microphone pair can lead to distortion of receiving pattern and degradation of the system performance. Thus, a calibration procedure is necessary.

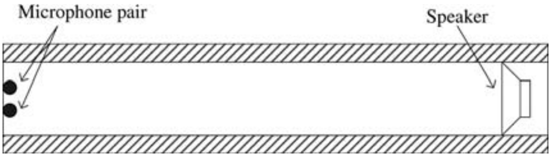


Fig. 2. Stationary wave pipe

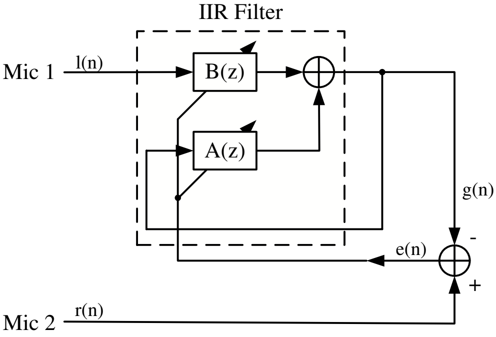


Fig. 3. Computation of calibration filter

Here we use a fixed 8th-order IIR filter to calibrate the differences of amplitude and phase between two microphones. Two microphones are placed closely in a stationary wave pipe as in Fig.2, while a speaker emits white Gaussian noise. Signals received by the two microphones are recorded simultaneously and then sent to a system shown in Fig.3 in which $l(n)$ and $r(n)$ are the received signals.

$$g(n) = \sum_{i=0}^{8} b_i(n)\, l(n-i) + \sum_{i=1}^{8} a_i(n)\, g(n-i) \tag{11}$$

In (11), $\{a_i(n), i = 1 \ldots 8\}$ and $\{b_i(n), i = 0 \ldots 8\}$ are coefficients of auto-regressive and moving-average portion at the moment of n, respectively.

$$e(n) = r(n) - g(n) \tag{12}$$

Eliminating the difference between the two microphones requires a minimum $e(n)$, which is accomplished with least mean square (LMS) algorithm using the output error method [8].

The IIR filter in the dashed line in Fig.3 converges to be the system we use to fulfill the calibration. The amplitude and phase of real transfer function of front microphone to back microphone before and after calibration is shown in Fig.4.

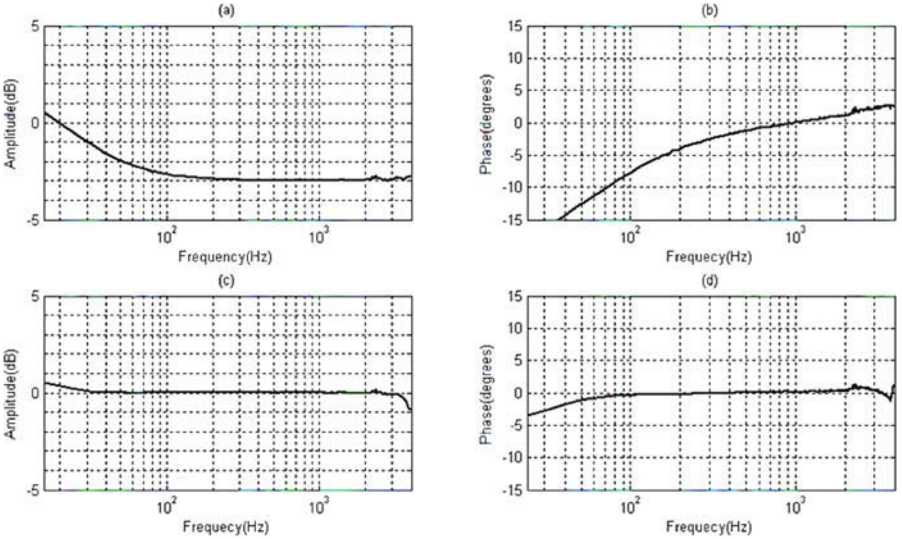


Fig. 4. Real transfer function of front mic to back mic. (a) and (b) are amplitude and phase of transfer function before calibration, respectively. (c) and (d) are those of after calibration. Frequency axes are shown in logarithmic form. Function value of (a) and (c) is in dB, while (b) and (d) in degree(°).

The real transfer function, which describes the degree of mismatch of the microphone pair, represents a system which is able to take $r(n)$ as input and get $l(n)$ at the output end. We neglect the system response with frequency lower than 100Hz because signal components within that range is of little significance as far as the application is concerned. Figure 4 shows that this calibration is very helpful to correct the distortion caused by microphone mismatch. Maximum distortions of phase and amplitude are reduced from 7.63° and 2.99dB to 1.15° and 0.27dB. This can be further shown in Sect.5 by the improvement of SNR.

4 Auditory Sub-band Null-Forming

From Sect.2, we learned that the relation between θ_{null} and $W(n)$ is monotonic. That is, one $W(n)$ decides only one θ_{null}. But considering the frequency cue we lose while simplifying (6) to get (7), this is not exactly the case. We can presume an example, when $d = 0.0425$m, $c = 340$m/s, $W(n) = -0.5$, and see θ_{null} vary at different frequency.

Freq.(Hz)	1000	2000	3000	4000
$\theta_{null}(°)$	110.75	114.39	124.88	167.63

That is to say, when the noise source comes from an angle of θ and we get a $W(n)$ adaptively, there is only one frequency point f_{opt} at which the receiving pattern has a null at θ. At other frequencies far away from f_{opt}, the null will deviate so that the system can not cancel these components as effectively as those near f_{opt}.

Furthermore, when the number of noise sources is more than one, the algorithm has a difficulty to steer the null to the right direction. θ_{null} will either converge between directions of the noise sources as an effect of average or vibrate between them. The noise reduction effect will thus be weakened.

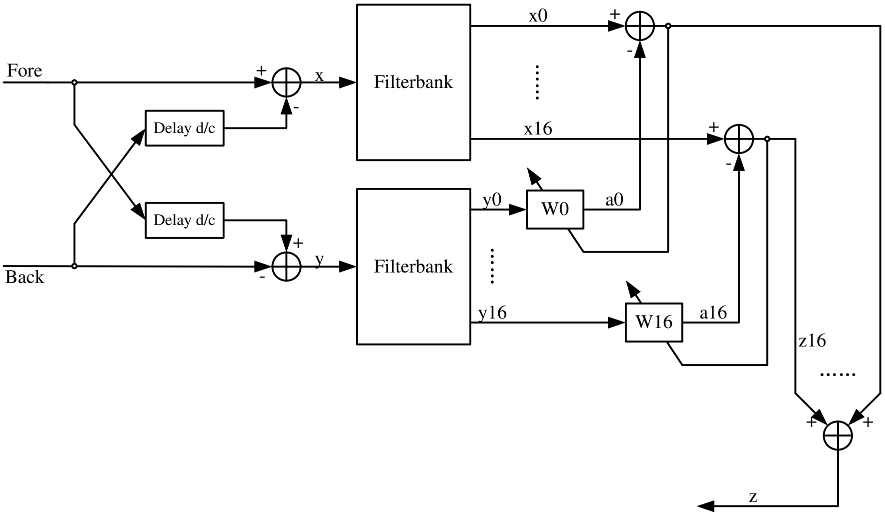


Fig. 5. Auditory Sub-band Null-Forming

To solve these problems, a system is developed as shown in Fig.5. Differential results of upper and lower branch are decomposed into 17 sub-bands respectively. Adaptive null-forming is implemented in each sub-band, after which the results are combined to make the final output. This enables the system to form a null separately in each band so the effect of deviation with frequency mentioned above is reduced. And angle of null in each band can be steered to different directions when under the circumstance of multiple interferences.

The sub-bands are made according to the Bark frequency group shown in Table 1 [9]:

Table 1. Bark Frequency Group

No.(Bark)	f_c/Hz	Δf/Hz	$10\log(\Delta f/1\text{Hz})$	f_l/Hz	f_h/Hz
1	50	80	19	20	100
2	150	100	20	100	200
3	250	100	20	200	300
4	350	100	20	300	400
5	450	110	20	400	510
6	570	120	21	510	630
7	700	140	21	630	770
8	840	150	22	770	920
9	1000	160	22	920	1080
10	1170	190	23	1080	1270
11	1370	210	23	1270	1480
12	1600	240	24	1480	1720
13	1850	280	25	1720	2000
14	2150	320	25	2000	2320
15	2500	380	26	2320	2700
16	2900	450	27	2700	3150
17	3400	550	27	3150	3700

fc, f_l and f_h indicate the central frequency, low boundary and high boundary of each band, respectively. Δf means the bandwidth $(f_h - f_l)$, and $10\log(\Delta f/1\text{Hz})$ is the relative bandwidth. (Our implementation uses a sampling frequency of 8000Hz so that the bands with frequency higher than 4000Hz are not listed.) The signal components within each group are judged integrally by brain. Thus the enhanced speech will sound more natural if division in frequency domain is made according to this biological basis.

Our experiments show that better auditory effect can be achieved by employing auditory sub-bands compared with some other sub-banding schemes. The filterbank for sub-banding features FIR filters with an order of 100 to provide frequency response with stop band adequately narrow. The processing can be optimized if polyphase filterbanks are used.

5 System Evaluation

5.1 Simulation Result

To test the performance of the proposed system, a computer based experiment is carried out in which a small room of 5m × 4m × 3m with a reverberant time of approximate 300ms is simulated using image method [10]. Two microphones are placed in the center of the room when speech and noise sources are assigned as Fig.6.

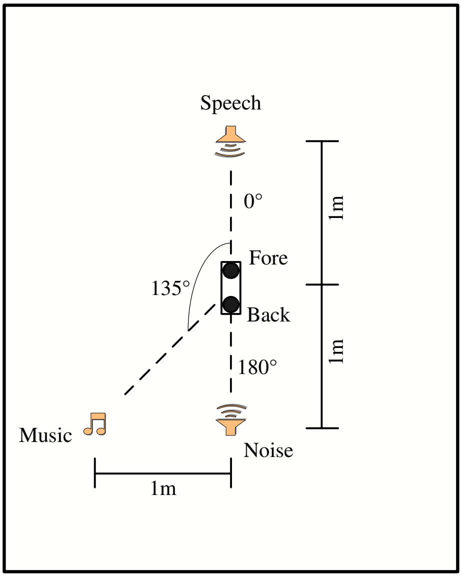


Fig. 6. Simulation experiment

Table 2. Performance comparison - Simulations (Unit:dB)

Group	Single Interference			Double Interferences		
SNR_{in}	-8.9	0.9	11.8	-10.8	-0.7	7.3
SNR^{ori}_{gain}	9.2	9.5	9.4	11.0	11.2	11.1
SNR^{sub}_{gain}	12.3	12.2	11.8	13.6	13.9	13.2

Speech source is placed in front of the array, at the direction of 0° and is 1m away from the microphones. White Gaussian noise is placed in the back, at 180° with the same distance away as speech. A non-stationary interference (music) is at 135°, 1m in the left of Gauss noise. The spacing between the two microphones is set to be 4.25cm.

In one group of the experiments, the music source is mute. There is only one source of noise under this circumstance. And in the other group, there are two interference sources. In both groups, signals are recorded at a sampling frequency of 8000Hz and with interferences in different intensities. All signals are simulated to be recorded ideally so calibration is not necessary here. The system performance is recorded in Table 2.

SNR_{in}, SNR^{ori}_{gain} and SNR^{sub}_{gain} indicate input SNR measured at one of the two microphones, improvement on SNR by original single-band null-forming algorithm and by sub-band method, respectively. An improvement of 2-3dB can be clearly observed.

5.2 Experiments in Anechoic Chamber

We also carry out the experiment in anechoic chamber. Assignment of instruments is the same as the simulation, shown in Fig.6. We select Knowles FG-3329 microphones to form the array. Two group of experiments are conducted, one with single interference and the other with two. The SNR of input signal is controlled to be around −5dB.

Table 3. Performance comparison - Anechoic chamber (Unit:dB)

Group	Single channel		Sub-band	
Calibration	No	Yes	No	Yes
Single Interference	4.6	10.7	6.0	12.2
Double Interferences	4.3	10.4	6.3	12.5

Table 3 shows the SNR gain of the original (single-channel) and sub-band method with or without calibration, under the noise condition of one or two interferences as depicted in Sect.5.1. An improvement of about 6dB is brought forward by calibration, while sub-banding contributes about 2dB enhancement.

6 Conclusions

We propose an improved adaptive null-forming algorithm using sub-band techniques with auditory features to increase the SNR gain as well as to preserve auditory effect. A fixed IIR filter was also included to calibrate microphone mismatch. Experiments show that these methods effectively improve the speech quality.

Future work may concern the combination of null-forming and spectral subtraction (SS), which is commonly used in single-channel speech enhancement. As shown in Fig.1, $y(n)$ could reasonably serve as an estimate of noise since speech signal contained in $y(n)$ is comparatively trivial if speaker stays near the direction of 0° of the array.

References

1. Manolakis, D.G., Ingle, V.K., Kogon, S.M.: Statistical and Adaptive Signal Processing. McGraw-Hill (2000)
2. Frost, O.L.: An algorithm for linearly constrained adaptive array processing. Proceedings of the IEEE **60**(8) (1972)
3. Griffths, L.J., Jim, C.W.: An alternative approach to linearly constrained adaptive beamforming. IEEE Trans. on Antennas and Propagation **30**(1) (1981)
4. Brandstein, M., Ward, D.: Microphone Arrays:Signal Processing Techniques and Applications. Springer (2001)
5. Zelinski, R.: A microphone array with adaptive post-filtering for noise reduction in reverberant rooms. IEEE Trans. on Signal Processing **52**(5) (2004)

6. Elko, G.W., Pong, A.T.N.: A simple adaptive first-order differential microphone. ASSP Workshop **15**(18) (1995)
7. Luo, F.L., Yang, J., Pavlovic, C., Nehorai, A.: Adaptive null-forming scheme in digital hearing aids. IEEE Trans. on Signal Processing **50**(7) (2002)
8. Haykin, S.: Adaptive Filter Theory. 4th edn. Prentice Hall (2002)
9. Yi, K., Tian, B., Fu, Q.: Speech Signal Processing. NDIP (2000)
10. Allen, J.B., Berkley, D.A.: Image method for efficiently simulating small-room acoustics. JASA **65**(4) (1979)

Multi-channel Noise Reduction in Noisy Environments

Junfeng Li[1], Masato Akagi[2], and Yôiti Suzuki[1]

[1] Research Institute of Electrical Communication, Tohoku University,
2-1-1, Katahira, Aoba-ku, Sendai, 980-8577, Japan
[2] School of Information Science, Japan Advanced Institute of Science and Technolgoy,
1-1, Asahidai, Nomi, Ishikawa, 923-1291, Japan
junfeng@ais.riec.tohoku.ac.jp, akagi@jaist.ac.jp,
yoh@ais.riec.tohoku.ac.jp

Abstract. Multi-channel noise reduction has been widely researched to reduce acoustic noise signals and to improve the performance of many speech applications in noisy environments. In this paper, we first introduce the state-of-the-art multi-channel noise reduction methods, especially beamforming based methods, and discuss their performance limitations. Subsequently, we present a multi-channel noise reduction system we are developing that consists of localized noise suppression by microphone array and non-localized noise suppression by post-filtering. Experimental results are also presented to show the benefits of our developed noise reduction system with respect to the traditional algorithms in terms of speech recognition rate. Some suggestions are finally presented for the further research.

Keywords: Multi-channel noise reduction, beamforming technique, localized noise, non-localized noise, speech recognition.

1 Introduction

Acoustic noise signals dramatically degrade the performance of many speech applications, such as speech communication system and automatic speech recognition system, in noisy environments [1]. For example, for speech communication system, acoustic noises degrade the quality and intelligibility of the received signals. For automatic speech recognition system, acoustic noises cause the mismatch between the training and testing conditions, further decreasing the recognition accuracy in real-world adverse conditions. Therefore, noise reduction must be very useful to improve the performance and robustness of these applications in noisy environments [1].

Though the problem of dealing with acoustic noises has been researched for several decades, it is currently still a challenging research topic. The challenges are mainly caused by the complex and time-varying characteristics of the signals (speech and noise signals) and acoustic environments [1], [2]. Desired speech signals have a broad-band and highly time-varying spectral components. In practical environments, noise signals have very complex and time-varying properties. Take the noise condition in a car environment as an example. Noises generated by winds around the car come from all directions and have slowly time-varying spectral components including coherent and incoherent noise components that are generally modeled as diffuse noise. Noises generated by engine come from certain directions and have

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 258 – 269, 2006.

slowly time-varying spectral components. Undesired interfering noises (e.g., passenger's voice and radio), however, have some determinable directions and highly non-stationary speech-like spectral components. Noises with different characteristics from various kinds of sources make it difficult to construct an effective noise reduction system. Furthermore, the characteristics of noises do vary with time and environments in an unpredictable fashion, further increasing the difficulty of designing a noise reduction system. Additionally, considering the practical implementation, real-time processing is generally a "must" for noise reduction systems in real conditions [1], [2].

To suppress various kinds of noises, many noise reduction algorithms have been published in the literature [1], [2]. Generally speaking, all of these noise reduction algorithms can be classified into two categories: single-channel technique and multi-channel technique, according to the number of microphones which are needed in the implementation.

A variety of single-channel noise reduction techniques [3], [4], [5], which exploit spectral and temporal differences between the speech and noise signals to suppress acoustical noises, have been proposed for speech enhancement and speech recognition. In real conditions, however, the speech and noise signals are considerably overlapped in the time-frequency domain, which makes it extremely difficult for single-channel techniques to substantially eliminate most of noise components without introducing speech distortion and artifacts (e.g., musical noise). As a result, single-channel techniques achieve very limited improvements in suppressing noise and in enhancing the speech enhancement and recognition performance [2].

In addition to the temporal and spectral characteristics, multi-channel techniques allow to exploit the spatial diversity of the speech and noise signals, resulting in the highly improved noise reduction performance [3]. In most scenarios, desired speech source and interfering noise source are physically located at different positions in the space. Exploiting the spatial diversity of the signals, multi-channel techniques can steer a main beam towards the desired speech source and/or nulls towards the interfering noise sources. Thus, compared to single-channel noise reduction techniques, multi-channel noise reduction techniques are substantially superior in suppressing the interfering signals arriving from the directions other than the specified "look" directions [2]. Additionally, among multi-channel noise reduction algorithms, post-filtering is normally needed to improve the entire performance in practical noisy environments. Therefore, multi-channel noise reduction systems with post-filtering have attracted increasing research interests [2].

In this paper, we first give a review of the state-of-the-art multi-channel (i.e., beamforming based) noise reduction systems ranging from the simple delay-and-sum beamformer to the advanced adaptive beamformers, as well as post-filtering. We then introduce the multi-channel noise reduction system we are developing consisting of localized noise suppression by microphone array and non-localized noise suppression by post-filtering. Experimental results are also presented to illustrate the benefits of our proposed system in terms of speech recognition accuracy in realistic environments where interfering signal and ambient noise are present. We finally provide some suggestions for the further research.

2 State-of-the-Art Multi-channel Noise Reduction

In comparison of single-channel noise reduction algorithms, multi-channel noise reduction algorithms have demonstrated a substantial superiority in reducing noise due to their spatial filtering capability. So far, many beamforming based algorithms have been reported in the literature [6], [7], [8], [9], [10], [11], [12], [13], [14]. The beamforming algorithms include fixed beamformer and adaptive beamformer, which are briefly discussed in the following sub-sections. Additionally, the widely used post-filtering algorithms are also discussed. Special attention is paid to the disadvantages of these existing algorithms.

2.1 Fixed Beamforming

The first class of beamforming techniques is fixed beamforming. In fixed beamforming techniques, the filter coefficients are normally optimized so that a beam is steered to the direction of the desired signal while suppressing the background noise coming from other directions as much as possible. These optimized filters are fixed, independent of the input signals, and then applied to the multi-channel microphone inputs [1], [2].

The simplest beamformer, referred to as delay-and-sum (DS) beamformer [2], [6], enhances the desired speech signal by summing the in-phase microphone signals after compensating for the arrival time differences of the desired sound signal to each microphone by inserting delays after each microphone, that is, the array is first electronically steered to the look-direction. In other words, in the DS beamforer, the weights of filters are fixed to for all frequencies and all frames. The advantages of the DS beamformer are that it is very simple to implement and that it minimizes the noise sensitivity and hence provides a high robustness against errors in the assumed signal model. However, a large number of microphones are normally needed to obtain an acceptable performance in real-world environments [2]. The superdirective beamformer is another widely studied fixed beamformer [7]. The superdirective beamformer maximizes the directivity index in the direction of the speech source for a diffuse noise field. Actually, the superdirective beamformer minimize the noise power of the beamformer output subject to distortionless response for the “look” direction, hence, it is also a *minimum variance distortionless response* (MVDR) beamformer. The implementation simplicity of the superdirective beamformer leads to its widely use in some known noise field. However, its data-independent property results in that only limited noise reduction performance can be obtained in practical time-varying environments [2].

Fixed beamforming techniques are widely used in the conditions where the acoustical characteristics do not change with time. However, using the fixed beamforming techniques, it is generally not possible to design arbitrary spatial directivity patterns for arbitrary microphone array configuration and design spatial directivity patterns which can be optimized to the time-varying acoustic environments [2], [7].

2.2 Adaptive Beamforming

The second class of beamforming techniques is adaptive beamforming. In contrast to fixed beamforming techniques, adaptive beamforming techniques make use of data-dependent filter coefficients that are adapted to respond to time-varying environments, yielding a better noise reduction performance than fixed beamforming techniques, particularly if the number of interference is small (i.e., smaller than the number of microphones) and in the acoustic environments with low reverberation [1], [2].

Adaptive beamforming techniques (e.g., the Frost beamformer) typically solve a *linearly constrained minimum variance* (LCMV) optimization problem [9], keeping the signals arriving from the desired look-direction (i.e., ideally the direction of the desired speech source) distortionless while suppressing the signals from other directions by minimizing the output power. The MVDR beamformer was proven as a special case of the LCMV beamformer under the assumption of zero correlations between the speech signal and the noise signal [2]. A *generalized sidelobe canceller* (GSC) beamformer, first presented by Griffiths and Jim as an alternative implementation structure of the LCMV beamformer, has also been widely researched [9]. The GSC beamformer consists of: a fixed beamformer which electronically steers the microphone array to the direction of interest (i.e., the speech source) and generates the so-called speech reference signal; a block matrix which steers the spatial nulls to the direction of speech source and generates the so-called noise reference signals; and a multi-channel noise canceller which suppress the residual noise components in the speech reference signal by using a multi-channel adaptive filter [9]. In addition, a wide variety of noise reduction algorithms that are based on the GSC beamformer have so far been suggested, which are of interest to be mentioned. Bitzer *et al.* presented an alternative implementation algorithm with a GSC structure of the superdirective beamforme and its performance was also analyzed in a diffuse noise field [10]. Fischer *et al.* proposed to apply a Wiener filter in the upper path of the GSC beamformer to suppress the uncorrelated noise components and then the correlated noise components are then reduced by the adaptive noise canceller in the lower path [11]. Recently, the GSC beamformer was extended to a *transfer function generalized sidelobe canceller* (TF-GSC) beamformer by considering the transfer functions which relate the speech source and the microphones, which was shown to yield high noise reduction performance in real-world environments [12]. Moreover, the theoretical performance of the GSC and TF-GSC beamformers was examined in the diffuse noise field [16], [17].

In all variants of the LCMV and GSC beamformers, adaptive signal processing (e.g., LMS) is normally used to avoid cancellation of the desired speech signal, which introduces low convergence rate in practical conditions and low ability in reducing non-stationary noise (e.g., sudden noise). Moreover, the adaptive beamformers only perform well and provide acceptable performance when the number of interfering noise sources is less than that of the microphones. Their performance will be greatly degraded by the reverberation effect and in the scenario where more noise sources exist (e.g., larger than the number of sensors) [2].

2.3 Post-Filtering

Multi-channel beamforming based algorithms provide high noise reduction performance especially for localized noise, however, only limited noise reduction performance is achieved in a diffuse noise field [2], [13], [14]. To further suppress residual noise at the beamformer output, post-filtering is normally needed to improve the noise reduction performance of the entire system in practical environments.

A variety of post-filtering techniques have been presented in the literature [13], [14] [15]. One commonly used multi-channel post-filter, which is based on Wiener filter, was first introduced by Zelinski [15]. The basic assumption behind this post-filter is that noises on different microphones are mutually uncorrelated, corresponding to a perfectly incoherent noise field. This assumption is, however, seldom satisfied in practical environments, especially for closely-spaced microphones and low frequencies, which are characteristics by the high-correlated noise [15].

To suppress the high-correlated noise, Fischer *et al.* proposed a noise reduction system which is based on the GSC beamformer [11]. The GSC beamformer suppresses the spatially coherent noise components, whereas a Wiener filter in the look direction is designed to suppress the spatially incoherent noise components [11]. However, Bitzer *et al.* pointed out that neither the GSC nor the standard Wiener post-filter performs well at low frequencies in a diffuse noise field [16], [17]. Therefore, they proposed to add a second post-filter at the output of a GSC beamformer with standard Wiener post-filter to reduce the spatially correlated noise components [18]. Recently, McCowan *et al.* developed a general expression of the Zelinski post-filter based on the a priori coherence function of the noise field [19]. Although this post-filter was shown to achieve improved speech quality and speech recognition accuracy compared to the Zelinski post-filter using the office room recordings, its performance is expected to be significantly degraded when difference between the “actual” and assumed coherence function exists.

3 Proposed Multi-channel Noise Reduction

3.1 Theoretical Principle of Proposed System --- Multi-channel Wiener Filter [2]

The underlying theoretical principle of our proposed multi-channel noise reduction system is the multi-channel Wiener filter, which provides an optimal solution to the problem of multi-channel noise reduction for broadband inputs in *minimum mean square error* (MMSE) sense [2]. With the assumption that the desired signal and noise signals are mutually uncorrelated, Simmer *et al.* showed that the multi-channel Wiener filter can further be decomposed into a MVDR beamformer followed by a single-channel Wiener post-filter [2]. As an extension of this algorithm, we propose a multi-channel noise reduction system consisting of localized noise suppression by a microphone array and non-localized noise suppression by post-filtering. The detailed description is given in the following subsections.

3.2 Signal Model in the Proposed System [14]

Let us consider an array of M microphones in a noisy environment. In our research, the observed signal on each microphone consists of three components. The first one is the desired speech signal $s(t)$ arriving from the direction such that the direction in arrival time between two microphones is ξ, The second is localized noise signals $n_p^c(t)$ arriving from the directions such that the time differences are $\delta_{m,p}\,(p=1,2,\ldots,P)$ and the third is non-localized signal $n^{uc}(t)$ which propagates in all directions simultaneously and is normally modeled as diffuse noise. Thus, the observed signal can further be represented as

$$x_m(t) = s(t - \xi_m) + \sum_{p=1}^{P} n_p^c(t - \delta_{m,p}) + n_m^{uc}(t) \tag{1}$$

Note that the localized noise signals $n_p^c(t)$ are generated by some point noise sources (e.g., fan, radio and competing speakers), which are fixed or movable in the space. Some localized noise sources are spectrally stationary or have slowly time-varying spectral properties (e.g., fan), while others are spectrally highly non-stationary (e.g., competing speech and sudden noise). The non-localized noise signals $n_m^{uc}(t)$ are generally modeled as diffuse noise (e.g., wind noise in car environments) arriving from all directions in the space. In most situations, these kinds of noise sources are spectrally stationary or have slowly time-varying spectral properties.

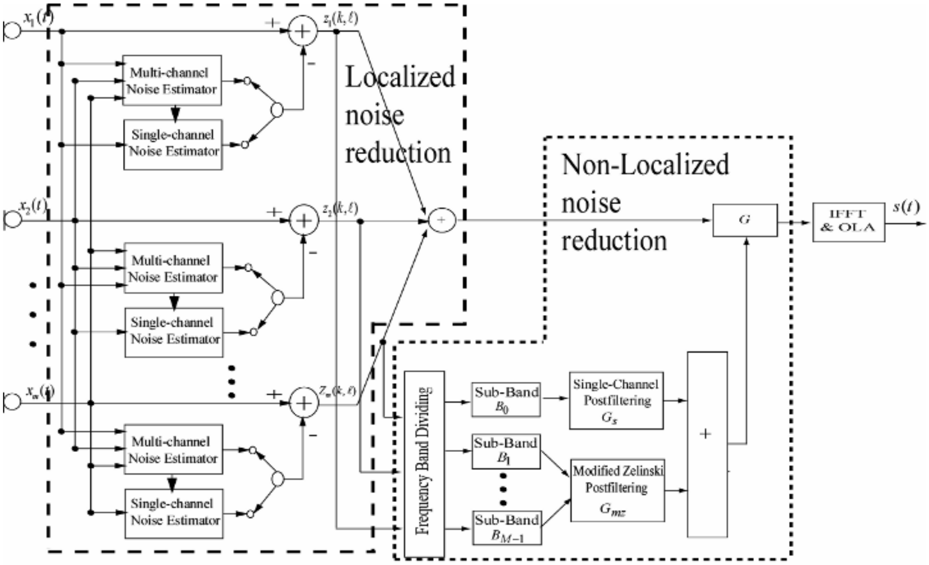

$x_1(t)$
$z_1(k,\ell)$
Multi-channel Noise Estimator
Single-channel Noise Estimator
Localized noise reduction
$x_2(t)$
$z_2(k,\ell)$
Non-Localized noise reduction
G
IFFT & OLA
$s(t)$
$x_m(t)$
$Z_m(k,\ell)$
Frequency Band Dividing
Sub-Band B_0
Sub-Band B_1
Sub-Band B_{M-1}
Single-Channel Postfiltering G_s
Modified Zelinski Postfiltering G_{mz}
+
-

Fig. 1. Block diagram of the proposed noise reduction system

3.3 Proposed Multi-channel Noise Reduction System

The objective of this research is to reduce both localized and non-localized noises while keeping the desired signal distortionless. In the proposed system, spectra of localized noises are first estimated using a hybrid noise estimation technique which combines a multi-channel approach and a single-channel approach and then subtracted from the spectra of noisy signals in each channel; non-localized noise is then reduced using a hybrid post-filter which is a Wiener filter in theory. The block diagram of the proposed noise reduction algorithm is shown in Fig. 1, including localized noise reduction and non-localized noise reduction.

3.3.1 Localized Noise Reduction [14], [20], [21]

To deal with localized noise components, we presented a microphone-array noise reduction algorithm based on a beamforming technique. The basic idea of our algorithm is that the spectra of localized noises are first estimated and then subtracted from those of the observed noise signals.

To accurately estimate the spectra of localized noise, we proposed a hybrid noise estimation technique in a parallel structure which combines a multi-channel estimation approach and a single-channel approach. The multi-channel estimation approach was implemented using the subtractive beamformer based method since it yields much more accurate spectral estimates for localized noises at most instances. The single-channel estimation approach was implemented using a soft-decision based noise estimation technique due to its ability in estimating the spectrum of non-stationary signal. Thus, the spectrum of localized noise in the *k*-th frequency bin and ℓ-th frame, $\hat{N}^c(k,\ell)$, calculated by this hybrid estimation technique, is given by [14], [20]

$$\hat{N}^c(k,\ell) = \begin{cases} \hat{N}_m^c(k,\ell), & \text{not array nulls,} \\ \hat{N}_s^c(k,\ell), & \text{array nulls,} \end{cases} \tag{2}$$

where $\hat{N}_m^c(k,\ell)$ and $\hat{N}_s^c(k,\ell)$ are the spectral estimates determined by the multi-channel approach [20] and the single-channel approach [22], respectively. The hybrid noise estimation technique is further enhanced by integrating a *robust and accurate speech absence probability* (RA-SAP) estimator [14]. Considering the strong correlation of speech presence uncertainty between adjacent frequency bins and consecutive frames, a RA-SAP estimator is developed. This RA-SAP estimator makes full use of the high estimation accuracy of the multi-channel estimation approach. Therefore, the final estimation accuracy for localized noises is greatly enhanced by the suggested RA-SAP estimator [14], [20]. The estimated spectra of localized noises are subsequently reduced from those of the noisy observations by using the non-linear spectral subtraction. More theoretically important, note that the subtractive beamformer based multi-channel localized noise suppression algorithm is in principle a MVDR beaformer [23].

3.3.2 Non-localized Noise Reduction [24]

At the output of localized noise reduction, the output signal $Z_m(k,\ell)$ on m-th channel, consisting of desired signal and beamformer-processed non-localized noise $D_m(k,\ell)$, is re-formulated the time-frequency domain as

$$Z_m(k,\ell) = S_m(k,\ell) + D_m(k,\ell). \tag{3}$$

Note that the non-localized noise component $D_m(k,\ell)$ is different from the non-localized component $N_m^{uc}(t)$ at the system input, since the localized noise reduction influences the non-localized components.

To further deal with the residual non-localized noise, we propose a Wiener post-filter with a hybrid structure under the assumption of a diffuse noise field. In the high frequency region, we present a modified Zelinski post-filter which considers and utilizes the correlation of noises on different microphones to improve the noise reduction with minimum speech distortion. The implementation of the modified Zelinski post-filter consists of four steps: determine the transient frequencies (i.e., the first minimum frequency of coherence function of diffuse noise field) according to the microphone array geometry; determine the microphone pairs on which noise is mutually uncorrelated for each frequency; compute the spectral densities of the desired and noisy signals; compute the gain function of the modified Zelinski post-filter. Finally, the gain function of the modified Zelinski post-filter is derived as [24]

$$G_{mz}(k,\ell)=\frac{\dfrac{1}{|\Omega_m(k)|}\sum\limits_{\{i,j\}\in\Omega_m(k)}\Re\left[\hat{\phi}_{Z_iZ_j}(k,\ell)\right]}{\dfrac{1}{|\Omega_m|}\sum\limits_{\{i,j\}\in\Omega_m(k)}\left[\dfrac{1}{2}\left(\phi_{Z_iZ_i}(k,\ell)+\phi_{Z_jZ_j}(k,\ell)\right)\right]}, \tag{4}$$

where Ω_m is the microphone pair sets for *m*-th sub-band on which noises are presumably low correlated, $\Re$ is the real part operation, $\hat{\phi}_{Z_iZ_j}$ and $\hat{\phi}_{Z_iZ_i}$ are the cross- and auto- spectral densities. Note that the first two steps can be done beforehand since they are only dependent on the microphone array geometry and independent of the input signals. Thus, the computational cost will greatly be reduced.

In the low frequency region, a single-channel technique is used to estimate the Wiener filter, given by [24]

$$G_s(k,\ell)=\frac{SNR_{\text{priori}}(k,\ell)}{1+SNR_{\text{priori}}(k,\ell)}, \tag{5}$$

where $SNR_{\text{priori}}(k,\ell)$ is the *a priori* SNR which is updated in a decision-directed scheme, significantly reducing the residual “musical noise” as detailed in [5]. More theoretically important, note that the proposed hybrid post-filter is in principle a Wiener filter [24].

4 Experiments and Results

We investigated the performance of the proposed noise reduction system using the speech enhancement experiments and the comprehensive speech recognition experiments. The noise reduction system was first performed on the multi-channel noisy signals, enhancing the desired speech signals. For the recognition experiments, these enhanced speech signals were further fed into the speech recognizer for recognizing the utterance. The performance improvements caused by the proposed noise reduction system (PRO-MAPF) were finally compared to those obtained by the traditional *delay-and-sum beamformer followed by Wiener post-filter* (DSWF) [11].

4.1 Speech Enhancement Experiments

To assess the performance of the proposed noise reduction algorithm, an equally-spaced linear array consisting of three microphones with the inter-element spacing of 10 cm was mounted in a car. The noise recordings were performed across all channels simultaneously at the sampling frequency of 12 kHz. The target signals and the interfering signals were the Chinese province/city names, uttered by one male and one female. The target speaker was placed in the front of the microphone array and the interfering speaker was placed with DOA of 60 degrees to the right. The integrated noise signals were first generated by mixing the car noise signals and the interfering signals at the same energy level. The observed noisy signals were created by adding the integrated noise signals into the target speech signals at 5 dB.

The speech enhancement results are plotted in Fig. 2. Fig. 2 (b) shows that the speech signals (北京，上海，广东，天津，重庆，内蒙古，宁夏，河南) were corrupted by both the interfering signals (广西，海南，四川，贵州，云南，西藏，香港) and the car noises. Fig. 2 (c) illustrates that the output of the DSWF is characterized by the high-level noise components (both the low-frequency car noises and the interfering signals). In contrast, the PRO-MAPF suppress almost all interfering signals and the car noises even in the regions where the speech and interfering signals are overlapped in the time-frequency domain, as shown in Fig. 2 (d). These results show that the PRO-MAPF is powerful in suppressing both localized and non-localized noise components.

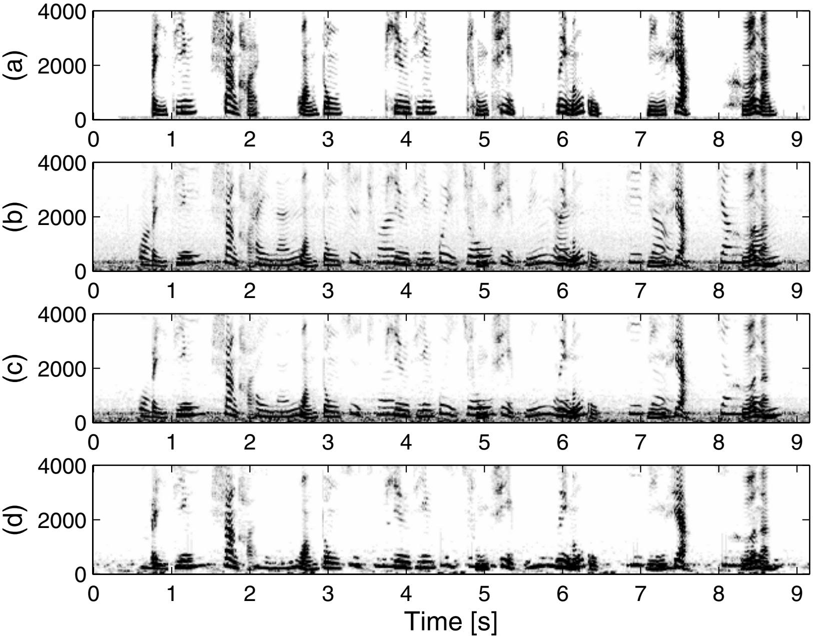


Fig. 2. Speech spectrograms. (a) Clean speech signal (北京，上海，广东，天津，重庆，内蒙古，宁夏，河南); (b) Noisy signal at the first microphone (SNR = 5 dB); (c) DSWF output; (d) PRO-MAPF output.

4.2 Speech Recognition Experiments

For speech recognition, the non-localized noises were the car noises same as used in speech enhancement experiments. The speech data were selected from AURORA-2J database for training and testing. For training, 8440 sentences uttered by 55 persons were used. For testing, two sets of noise-corrupted data were generated. The first data set (*Set A*) involved the addition of the car noise recordings and 1001 test sentences at different SNRs from 0 to 20dB with 5dB step. The second data set (*Set B*) involved the addition of the multi-channel car noises and a passenger's voice which was Japanese digit /iti/ with DOA of 60 degrees to the right, across 1001 test sentences at the different SNRs same as in *Set A*. Note that *Set B* corresponds to a realistic context for a typical car condition where a passenger is speaking.

4.3 Experimental Results

The recognition results for the noise reduction systems (DSWF and PRO-MAPF) in two noise conditions (*Set A* and *Set B*) are presented in Fig. 3.

As Fig. 3 (left) shows, for data *Set A*, all tested noise reduction algorithms provide some degree of performance improvement in speech recognition rate compared with noisy inputs. The average recognition rate improvement achieved by DSWF algorithm amounts to 6.0% with respect to noisy inputs. Whereas, the highest recognition rate improvement of about 18.6% was achieved by our PRO-MAPF. The recognition rate improvements drastically increase as the noise level increase. Moreover, in very high SNR conditions, all the tested algorithms provide just slight performance improvement compared with the noisy inputs, which is reasonable since the inputs are "clean" enough and a relatively high recognition rate is obtained in these conditions.

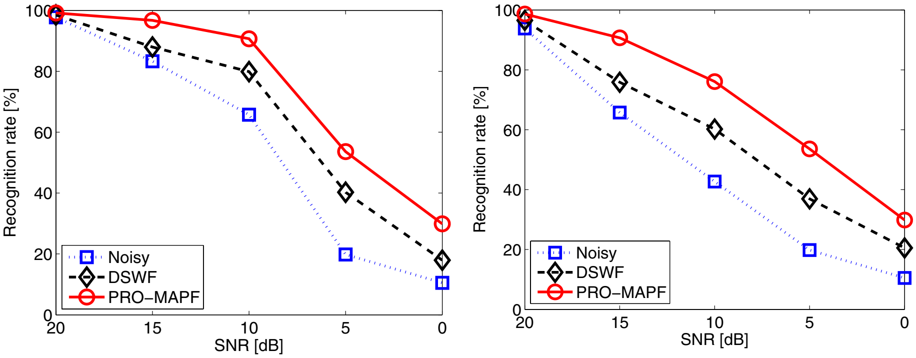


Fig. 3. Speech recognition results for the testing data *Set A* (left) and for the testing data *set B* (right)

Concerning the recognition results for data *Set B* shown in Fig. 3 (right), we can observe that PRO-MAPF also demonstrates highest recognition rate at all SNRs. In this noise condition, the recognition rate goes down greatly for unprocessed noisy testing data. Recognition rate improvements of 11.5% and 23.2% were demonstrated by the DSWF and PRO-MAPF algorithms. The highest recognition rate of PRO-MAPF can be attributed to the fact that it is successful in dealing with both passenger's interfering speech and diffuse car noise simultaneously with minimum speech distortion, resulting in the higher speech recognition rate.

5 Suggestions for Further Research

In this research, we have so far developed a noise reduction algorithm that is designed using microphone array and post-filtering in noisy environments. Its performance was evaluated in various car noise conditions and was further shown to outperform many traditional noise reduction algorithms in terms of speech recognition rate. However, the proposed noise reduction algorithm should be further improved in the following ways. (1) So far, the input microphone signals were assumed to be perfectly time-aligned in advance, that is, the desired speech signals were assumed to come from the front of the microphone array. In the practical implementation, it is necessary to take into account of the transfer function between the desired speech source and microphones. (2) Because of the small-size microphone array, improving the robustness of the noise reduction system against imperfections, such as the imperfection of microphone positions, is necessary for the real-world implementation, which is suggested as well for further research. (3) Moreover, in the real-world environments, the performance degradation of hands-free speech recognition systems is caused by not only acoustic background noise, but also reverberation and acoustic echoes. To further improve the performance of many speech applications in practical conditions, it is necessary to further deal with reverberation and acoustic echoes by combining the proposed noise reduction algorithm with other advanced dereverberation and echo cancellation techniques.

Reference

1. J. Benesty, S. Makino and J. Chen (eds.), Speech Enhancement, Springer, Verlag, 2005.
2. M. S. Brandstein and D. B. Ward (eds.), Microphone Arrays: Signal Processing Techniques andApplications, Springer-Verlag, 2001.
3. S. F. Boll. "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. on ASSP,,* vol. ASSP-28, no. 2, pp. 113-120, 1979.
4. Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans ASSP,* vol. 32, pp. 1109-1121, 1984.
5. -----, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. on ASSP,* vol. 33, no. 2, pp. 443-445, 1985.
6. K. U. Simmer and A. Wasiljeff, "Adaptive microphone arrays for noise suppression in the frequency domain," In *Proc. WAAC*, 1992.
7. H. Cox, R. M. Zeskind and M. M. Owen, " Robust adaptive beamforming," *IEEE on ASSP*, vol. ASSP-35, no. 10, pp. 1365-1375, 1987.
8. O. L. Frost, "An algorithm for linearly constrained adaptive array processing," *In Proc. Of the IEEE*, vol. 60, no. 8, pp. 926-935, 1972.
9. L. J. Griffiths and C. W. Jim, "An alternative approach to linear constrained adaptive beamforming," *IEEE Trans. on Antennas and Propagation,* vol. AP-30, pp. 27-34, 1982.

10. J. Bitzer, K. D. Kammeyer, and K. U. Simmer, "An alternative implementation of the superdirective beamformer," *In Proc. IEEE WASSPAA1999,* pp. 1-4, 1999.
11. S. Fischer and K. D. Kammeyer, "Beamforming microphone arrays for speech acquisition in noisy environments," *Speech Communication*, pp.215-227, 1996.
12. S. Gannot, D. Burshtein, and E. Weinstein "Signal enhancement using beamforming and nonstationary with applications to speech," *IEEE Trans. on Signal Processing,* vol. 49, no. 8, pp. 1614-1626, 2001.
13. I. Cohen, "Multi-channel post-filtering in non-stationary noise environments," *IEEE Trans. on Signal Processing,* vol. 52, no. 5, pp. 1149-1160, 2004.
14. J. Li and M. Akagi, "A noise reduction system based on hybrid noise estimation technique and post-filtering in arbitrary noise environments," *Speech Communication*, vol. 48, no. 2, pp. 111-126, 2006.
15. R. Zelinski, "A microphone array with adaptive post-filtering for noise reduction in reverberant rooms," *In Proc. ICASSP,* pp. 2578-2581, 1988.
16. J. Bitzer, K. U. Simmer and K. D. Kammeyer, "Multichannel noise reduction: Algorithm and Theoretical limits," *In Proc. ESPC*, pp. 105-108, 1998.
17. ----- , "Theoretical noise reduction limits of the generalized sidelobe canceller for speech enhancement," *In Proc. ICASSP*, pp. 100-103, 1999.
18. J. Bitzer, K. U. Simmer and K. D. Kammeyer, "Multi-microphone noise reduction by post-filter and superdirective beamformer," *In Proc. IWAENC*, pp. 100-103, 1999.
19. I. A. McCowan and H. Bourland, "Microphone array post-filter based on noise field coherence," *IEEE Trans. on Speech and Audio Processing,* vol. 11, pp. 709-716, 2003.
20. J. Li and M. Akagi, "Noise reduction using hybrid noise estimation techniques and post-filtering," *In Proc. ICSLP*, pp. 2705-2708, 2004.
21. -----, X. Lu and M. Akagi, "A noise reduction system in arbitrary noise environments and its applications to speech enhancement and speech recognition," *In Proc. ICASSP*, pp. 277-280, 2005.
22. I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Processing*, vol. 81, pp. 2403-2418, 2001.
23. J. Li and M. Akagi, "Noise reduction based on generalized subtractive beamformer," *Acoustical Science and Technology*, vol. 27, no. 4, 2006.
24. -----, "A hybrid microphone array post-filter in a diffuse noise field," *In Proc. Eurospeech*, pp. 2313-2316, 2005.

Minimum Phone Error (MPE) Model and Feature Training on Mandarin Broadcast News Task

Jia-Yu Chen[1], Chia-Yu Wan[1], Yi Chen[1], Berlin Chen[2], and Lin-shan Lee[1]

[1] Graduate Institute of Communication Engineering, National Taiwan University, Taipei
[2] Graduate Institute of Computer Science and Information Engineering, National Taiwan Normal University, Taipei
chiayu.chen@gmail.com, chiayui@speech.ee.ntu.edu.tw, chenyi@speech.ee.ntu.edu.tw, berlin@csie.ntnu.edu.tw, lslee@gate.sinica.edu.tw

Abstract. The Minimum Phone Error (MPE) criterion for discriminative training was shown to be able to offer acoustic models with significantly improved performance. This concept was then further extended to Feature-space Minimum Phone Error (fMPE) and offset fMPE for training feature parameters as well. This paper reviews the concept of MPE and reports the experiments and results in performing MPE, fMPE and offset fMPE on the task of Mandarin Broadcast News, and significant improvements were obtained similar to the results reported for other languages and other tasks by other sites. In addition, a new concept of dimension-weighted offset fMPE is proposed in this work and even better performance than offset fMPE was obtained.

Keywords: Discriminative training, minimum phone error (MPE), feature-space MPE (fMPE), offset fMPE, dimension-weighted, large-vocabulary continuous speech recognition (LVCSR).

1 Introduction

The MPE (Minimum Phone Error) criterion for discriminative training has been shown to offer HMM parameters with significantly improved performance [1, 2]. The basic concept of MPE is similar to other discriminative training approaches, in which the acoustic models are trained in such a way to force the acoustic models to recognize the training data correctly, or to try to differentiate the acoustic models when they turned out to be confusing with respect to the training data.

Feature-space MPE (fMPE) is an extension of MPE in order to obtain improved version of feature parameters rather than acoustic models [3]. The basic idea is to train a feature-level transformation which offsets the original features to an optimal set of features, and the transformation is defined by a linear (matrix) projection from high-dimensional feature space based on posterior probabilities of Gaussians. Offset fMPE is then a further extension of fMPE [4].

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 270 – 281, 2006.

This paper reviews the concept of MPE and reports the experiments and results in performing MPE, fMPE and offset fMPE on the task of Mandarin Broadcast News, and significant improvements were obtained similar to the results reported for other languages and other tasks by other sites. In addition, a new concept of dimension-weighted offset fMPE is proposed in this work, and even better performance than offset fMPE was obtained.

This paper is organized as follows. In section 2, we briefly review several important discriminative training criteria derived from the basic minimum Bayesian risk principle. Section 3 and 4 then describe the frameworks of MPE and fMPE in more detail. In section 5, the dimension-weighted offset fMPE proposed in this paper is compared with fMPE and offset fMPE. Section 6 presents the experimental results, and section 7 gives the conclusions.

2 Discriminative Training Criteria and MPE

The minimum Bayesian risk (MBR) criterion is defined as

$$\begin{aligned}(\lambda,\Gamma) &= \arg\min_{\lambda',\Gamma'} \sum_r R(s_r \mid O_r) \\ &= \arg\min_{\lambda',\Gamma'} \sum_r \sum_u P(u \mid O_r) L(u, s_r)\end{aligned} \tag{1}$$

where O_r is the r-th training utterance, λ is the acoustic model, Γ is the language models, s_r and u represents the correct transcriptions and the recognized results respectively. $L(u, s_r)$ is the loss function caused by the difference between u and s_r, and $P(u \mid O_r)$ is the posterior probability of u. The discriminative training is to estimate model parameters which minimize the above Bayesian risk. The reduction of Bayesian risk then leads to the reduction of word error rate. Quite several training criteria are extensions of this minimum Baysian risk criterion, as will be briefly summarized below.

When the loss function is defined as a zero-one function,

$$L(u, s_r) = \begin{cases} 0, & u = s_r \\ 1, & u \neq s_r, \end{cases} \tag{2}$$

equation (1) is reduced to

$$(\lambda,\Gamma) = \arg\max_{\lambda',\Gamma'} \sum_r p(s_r \mid O_r) \tag{3}$$

$$= \arg\max_{\lambda',\Gamma'} \sum_r \log \frac{p(O_r \mid s_r) p(s_r)}{p(O_r)} . \tag{4}$$

Equation (3) is the criterion of MAP. If we assume that the prior probability $p(O_r)$ is a uniform distribution, equation (4) can be reduced to

$$(\lambda, \Gamma) = \arg\max_{\lambda', \Gamma'} \sum_r \log p(O_r \mid s_r) p(s_r) \tag{5}$$

which is the objective function of maximum likelihood estimation (MLE).

If $p(O_r)$ is expressed as $p(O_r) = \sum_v p(O_r \mid v) p(v)$ where v is one of the possible recognition results, we have

$$(\lambda, \Gamma) = \arg\max_{\lambda', \Gamma'} \sum_r \log \frac{p(O_r \mid s_r) p(s_r)}{\sum_v p(O_r \mid v) p(v)} \tag{6}$$

which is the objective function of maximum mutual information (MMI) training.

On the other hand, overall risk criterion estimation (ORCE) minimizes the Bayesian risk directly. The loss function is defined as the Levenhstein distance between the N-best and the correct transcription while focusing on word error rate reduction,

$$L(u, s_r) = \begin{cases} 0, & u = s_r \\ e(u, s_r), & u \neq s_r, \end{cases} \tag{7}$$

and the training criterion becomes the criterion for ORCE,

$$(\lambda, \Gamma) = \arg\min_{\lambda', \Gamma'} \sum_r \sum_u \frac{p(O_r \mid u) p(u)}{\sum_v p(O_r \mid v) p(v)} e(u, s_r). \tag{8}$$

Minimum phone error (MPE) training aims to maximize the expected phone accuracy,

$$\begin{aligned} (\lambda, \Gamma) &= \arg\max_{\lambda', \Gamma'} \sum_r \sum_u p(u \mid O_r) Acc(u, s_r) \\ &= \arg\max_{\lambda', \Gamma'} \sum_r \sum_u \frac{p(O_r \mid u) p(u)}{\sum_v p(O_r \mid v) p(v)} Acc(u, s_r) \end{aligned} \tag{9}$$

where the accuracy of the recognized result u, $Acc(u, s_r)$, weighted by its posterior probability is the target of maximization, and all possible recognized results are simulated by the word graph. MPE in equation (9) are actually quite different from ORCE in equation (8). The former focuses on phone accuracy and is implemented on a word graph, while the latter focuses on word error rate and is implemented on N-best results. Besides, MPE introduces the prior distribution of the new estimated models which are not taken into consideration by ORCE.

MPE can also be regarded as a variation of MMI. MMI treated the correct transcriptions as the numerator lattice and the whole word graph as the denominator lattice or the competing sequences, while MPE treats all possible correct sequences on the word graph as the numerator lattice, and treats all possible wrong sequences as the denominator lattice. From this point of view, MPE takes more advantages of the word graph than MMI.

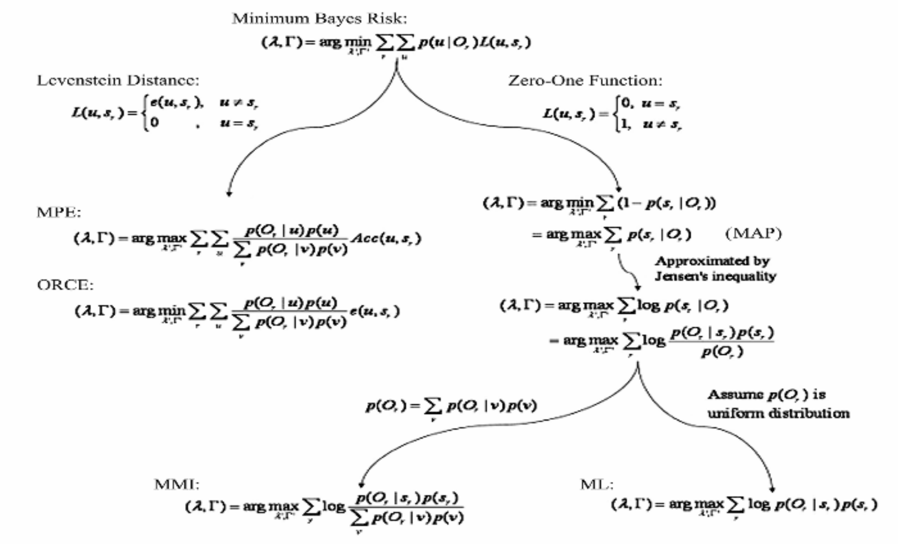

Minimum Bayes Risk:
$(\lambda,\Gamma)=\arg\min_{\lambda;\Gamma}\sum_r\sum_u p(u|O_r)L(u,s_r)$
Levenstein Distance:
$L(u,s_r)=\begin{cases}e(u,s_r), & u\neq s_r\\ 0, & u=s_r\end{cases}$
Zero-One Function:
$L(u,s_r)=\begin{cases}0, & u=s_r\\ 1, & u\neq s_r\end{cases}$
MPE:
$(\lambda,\Gamma)=\arg\max_{\lambda;\Gamma}\sum_r\sum_u\frac{p(O_r|u)p(u)}{\sum_v p(O_r|v)p(v)}Acc(u,s_r)$
ORCE:
$(\lambda,\Gamma)=\arg\min_{\lambda;\Gamma}\sum_r\sum_u\frac{p(O_r|u)p(u)}{\sum_v p(O_r|v)p(v)}e(u,s_r)$
$(\lambda,\Gamma)=\arg\min_{\lambda;\Gamma}\sum_r(1-p(s_r|O_r))$
$=\arg\max_{\lambda;\Gamma}\sum_r p(s_r|O_r)$ (MAP)
Approximated by Jensen's inequality
$(\lambda,\Gamma)=\arg\max_{\lambda;\Gamma}\sum_r\log p(s_r|O_r)$
$=\arg\max_{\lambda;\Gamma}\sum_r\log\frac{p(O_r|s_r)p(s_r)}{p(O_r)}$
$p(O_r)=\sum_v p(O_r|v)p(v)$
Assume $p(O_r)$ is uniform distribution
MMI:
$(\lambda,\Gamma)=\arg\max_{\lambda;\Gamma}\sum_r\log\frac{p(O_r|s_r)p(s_r)}{\sum_v p(O_r|v)p(v)}$
ML:
$(\lambda,\Gamma)=\arg\max_{\lambda;\Gamma}\sum_r\log p(O_r|s_r)p(s_r)$

Fig. 1. The derivation flow of the various training criteria

Fig. 1 shows the derivation flow of the various training criteria reviewed above.

3 Minimum Phone Error (MPE)

The objective function of equation (9) is

$$F_{MPE}(\lambda)=\sum_r\sum_u P(u|O_r)Acc(u,s_r)=\sum_r\sum_u\frac{P(O_r|u)P(u)}{\sum_{v\in lat}P(O_r|v)P(v)}Acc(u,s_r) \qquad (10)$$

where $Acc(u,s_r)$ is the accuracy of u, v is one out of all possible recognized results simulated by the word graph, and $p(u|O_r)=\frac{p(O_r|u)p(u)}{\sum_v p(O_r|v)p(v)}$ is the posterior of u.

If we replace equation (10) by an auxiliary function and add it to a smoothing function, the training criterion becomes,

$$\begin{aligned}G_{MPE}(\lambda,\lambda') &=\sum_r\sum_{q\in lat}\sum_{t=s_q}^{e_q}\sum_s\sum_m\gamma_q^{MPE}\gamma_{qsm}(t)\log N(O(t),\mu_{sm},\Sigma_{sm})\\ &-\sum_s\sum_m\frac{D_{sm}}{2}[\log(2\pi)^d+\log(|\Sigma_{sm}|)+(\mu_{sm}-\mu'_{sm})^T\Sigma_{sm}^{-1}(\mu_{sm}-\mu'_{sm})+tr(\Sigma'_{sm}\Sigma_{sm}^{-1})]\end{aligned} \qquad (11)$$

where $\gamma_{qsm}(t)$ is the occupation probability of state s, mixture m, arc q at time t, μ'_{sm} and Σ'_{sm} are current model parameters, μ_{sm} and Σ_{sm} are the new estimated model parameters, and $\gamma_q^{MPE} = \dfrac{\partial F_{MPE}(\lambda)}{\partial \log p(q)} = \gamma_q (C_q - C_{avg})$, where $\gamma_q = \dfrac{\sum_{s:\, q\in s} p(O_r \mid s)p(s)}{\sum_v p(O_r \mid v)p(v)}$ is the posterior probability of word arc q , $C_q = \dfrac{\sum_{u:\, q\in u} p(O_r \mid u)p(u)Acc(u, s_r)}{\sum_{q\in s} p(O_r \mid s)p(s)}$ is the expected phone accuracy of all paths passing through word arc q, and $C_{avg} = \dfrac{\sum_u p(O_r \mid u)p(u)Acc(u, s_r)}{\sum_v p(O_r \mid v)p(v)}$ represents the expected phone accuracy of all sequences on the word graph. For an arc q, if $C_q > C_{avg}$, q will be classified as in the numerator lattice and used in positive training; if $C_q < C_{avg}$, q will be classified as in the denominator lattice and used in negative training. If $C_q = C_{avg}$, this means q has no competitors, or all other competitors have the same model as q .

The statistics required for MPE training are computed from the numerator and the dominator lattices,

$$\gamma_{sm}^{num} = \sum_r \sum_{q\in lat} \sum_{t=s_q}^{e_q} \max(0, \gamma_q^{MPE})\gamma_{qsm}(t) \tag{12}$$

$$\gamma_{sm}^{den} = \sum_r \sum_{q\in lat} \sum_{t=s_q}^{e_q} \max(0, -\gamma_q^{MPE})\gamma_{qsm}(t) \tag{13}$$

$$\theta_{sm}^{num}(O) = \sum_r \sum_{q\in lat} \sum_{t=s_q}^{e_q} \max(0, \gamma_q^{MPE})\gamma_{qsm}(t)\ O_r(t) \tag{14}$$

$$\theta_{sm}^{den}(O) = \sum_r \sum_{q\in lat} \sum_{t=s_q}^{e_q} \max(0, -\gamma_q^{MPE})\gamma_{qsm}(t)\ O_r(t) \tag{15}$$

$$\theta_{sm}^{num}(O^2) = \sum_r \sum_{q\in lat} \sum_{t=s_q}^{e_q} \max(0, \gamma_q^{MPE})\gamma_{qsm}(t)\ O_r(t)O_r^T(t) \tag{16}$$

$$\theta_{sm}^{den}(O^2) = \sum_r \sum_{q\in lat} \sum_{t=s_q}^{e_q} \max(0, -\gamma_q^{MPE})\gamma_{qsm}(t)\ O_r(t)O_r^T(t) \tag{17}$$

From the differential of equation (11), the model update equations are,

$$\mu_{sm} = \frac{\theta_{sm}^{num}(O) - \theta_{sm}^{den}(O) + D_{sm}\mu_{sm}^{'}}{\gamma_{sm}^{num} - \gamma_{sm}^{den} + D_{sm}} \tag{18}$$

$$\Sigma_{sm} = \frac{\theta_{sm}^{num}(O^2) - \theta_{sm}^{den}(O^2) + D_{sm}(\mu_{sm}^{'}\mu_{sm}^{'T} + \Sigma_{sm}^{'T})}{\gamma_{sm}^{num} - \gamma_{sm}^{den} + D_{sm}} - \mu_{sm}\mu_{sm}^{T} \tag{19}$$

The update equations try to make the new model parameters closer to the features of the numerator lattice and farther away from those of the denominator lattice.

4 Feature-Space MPE (fMPE)

4.1 fMPE

Feature-space minimum phone error (fMPE) is a discriminative training method which adds an offset to the old feature,

$$y_t = x_t + Mh_t \tag{20}$$

where x_t is the old feature and y_t is the new feature. M is a transform matrix estimated by MPE and updated by gradient descent. h_t is composed of 100000 Gaussian posterior probabilities spliced by adjacent frames [3].

For a certain element M_{ij} of M , it can be updated as

$$M_{ij} = M_{ij} + v_{ij}\frac{\partial F}{\partial M_{ij}} \tag{21}$$

$$\frac{\partial F}{\partial M_{ij}} = \sum_{t=1}^{T}\frac{\partial F}{\partial y_{ti}}\frac{\partial y_{ti}}{\partial M_{ij}} = \sum_{t=1}^{T}\frac{\partial F}{\partial y_{ti}}h_{tj} \tag{22}$$

When using only direct differential to update the features, as shown in equation (23), significant improvements are obtainable but then lost very soon when the acoustic model is retrained with ML. The indirect differential part thus aims to reflect the model change from the ML training with new features, as shown in equation (24). Therefore, the features and models can be trained iteratively. Feature differential is then the sum of direct differential and indirect differential, as shown in equation (25):

$$\frac{\partial F}{\partial y_{ti}}^{direct} = \sum_{s=1}^{S}\sum_{m=1}^{M}\frac{\partial F}{\partial l_{smt}}\frac{\partial l_{smt}}{\partial y_{ti}} \tag{23}$$

$$\frac{\partial F}{\partial y_{ti}}^{indirect} = \sum_{s=1}^{S}\sum_{m=1}^{M}\frac{\gamma_{sm}(t)}{\gamma_{sm}}(\frac{\partial F}{\partial \mu_{smi}} + 2\frac{\partial F}{\partial \sigma_{smi}^{2}}(y_{ti} - \mu_{smi})) \tag{24}$$

$$\frac{\partial F}{\partial y_{ti}} = \frac{\partial F}{\partial y_{ti}}^{direct} + \frac{\partial F}{\partial y_{ti}}^{indirect} \tag{25}$$

In the above equations, l_{smt} is the log likelihood of state s, mixture m at time t, $\gamma_{sm}(t)$ is the occupation probability of state s and mixture m at time t, and γ_{sm} is the sum of $\gamma_{sm}(t)$ over t, or $\gamma_{sm} = \sum_{t=1}^{T} \gamma_{sm}(t)$.

4.2 Offset fMPE

The difference of offset fMPE from the original fMPE is the definition of the high-dimensional vector h_t of posterior probabilities [4]. For offset fMPE, h_t is defined as

$$\begin{aligned} h_t = [5.0\gamma_t^1,\ & \gamma_t^1(x_t(1)-\mu^1(1))/\sigma^1(1),\ \gamma_t^1(x_t(2)-\mu^1(2))/\sigma^1(2),... \\ & 5.0\gamma_t^2,\ \gamma_t^2(x_t(1)-\mu^2(1))/\sigma^2(1),\ \gamma_t^2(x_t(2)-\mu^2(2))/\sigma^2(2),...] \end{aligned} \tag{26}$$

where γ_t^i represents the posterior of i-th Gaussian at time t, $\gamma_t^i = \frac{p(O_t \mid g_i)}{\sum_{j=1}^{N} p(O_t \mid g_j)}$, and g_j is the j-th Gaussian. Offset fMPE uses the Gaussian posterior probability and the offset (i.e., feature vector subtracts the mean and divided by the standard deviation, then weighted by the Gaussian posterior probability). The Gaussian posterior probability is given higher weight (5.0) as it is much more important [4]. The number of Gaussians needed is about 1000, which is significantly lower than 100000 for the original fMPE.

5 Dimension-Weighted Offset fMPE Proposed in This Paper

Different from the offset fMPE which gives the same weight on each dimension of the feature offset vector, the dimension-weighted offset fMPE proposed in this paper calculates the posterior probability on each dimension of the feature offset vector, $\gamma_t^i(d) = \frac{p(O_t(d) \mid g_i(d))}{\sum_{j=1}^{N} p(O_t(d) \mid g_j(d))}$, where $g_i(d)$ is the i-th Gaussian on dimension d. When the Gaussian likelihood $p(O_t(d) \mid g_i(d))$ is calculated, normalization is performed on each dimension with $\gamma_t^i(d)$,

$$\begin{aligned} h_t = [&5.0\gamma_t^1,\ \ \gamma_t^1(1)(x_t(1)-\mu^1(1))/\sigma^1(1),\ \ \gamma_t^1(2)(x_t(2)-\mu^1(2))/\sigma^1(2),\ldots \\ &5.0\gamma_t^2,\ \ \gamma_t^2(1)(x_t(1)-\mu^2(1))/\sigma^2(1),\ \ \gamma_t^2(2)(x_t(2)-\mu^2(2))/\sigma^2(2),\ldots] \end{aligned} \tag{27}$$

So the weight is different according to the posterior probability of each dimension.

6 Experimental Results

6.1 Experimental Setup

Two sets of feature parameters were tested in the large vocabulary Mandarin speech recognition experiments: (1) MFCC features: the conventional 39-dimensional MFCC feature vectors consist of 12 MFCCs and the log energy, and their first and second derivatives. Utterance-based Cepstral Mean Subtraction (CMS) was applied to all the training and testing materials. (2) HLDA features: Heteroscedastic Linear Discriminant Analysis (HLDA) [6, 7] was directly applied on the Mel-scale filterbank outputs to construct the 39-dimensional feature vectors. Maximum Likelihood Linear Transform (MLLT) [8] and Cepstral Normalization (CN) were then applied on these feature vectors.

The speech corpus for training and testing is from the Mandarin Broadcast News corpus (MATBN) collected in Taipei [9]. About 25 hours of gender-balanced data for the field reporters collected between Nov 2001 and Dec 2002 were used for training, while another set of 1.5-hour data of field reporters collected within 2003 for testing. The baseline acoustic models for both MFCC and HLDA features were trained by 40 iterations of ML training.

The lexicon size used for this task is 72K words. The background language model is trained on the Chinese News Agency (CNA) 2001 and 2002 text corpus, including roughly 170 million characters. Trigram models were used. Meanwhile the reference transcriptions of the 25-hour training utterances, consisting of about 500K characters, were regarded as in-domain text corpus and used to train an in-domain language model, which is then to be interpolated with the background language model and used as the final language model for the experiments.

6.2 MPE Model Training

The results for MPE model training with 10 iterations are listed in the center row of Table 1. It can be found that the MPE model training significantly improved the performance on syllable, character and word levels regardless of the features used. For MFCC features, the absolute error rate reduction was 3.51% in syllable error rate (SER) (21.04% to 17.53%), 3.64% in character error rate (CER) (28.53% to 24.89%), and 3.90% in word error rate (WER) (37.33% to 33.43%). Similar results were obtained for HLDA features, for example with an absolute CER reduction of 2.81% (25.52% to 22.71%), or 11.01% relative CER reduction. Note that although HLDA features have been transformed discriminatively, the MPE training here was still able to give significant extra improvements. Also, as shown in Fig. 2(a), (b) and (c), the performance of MPE converged at 6-9 iterations for MFCC features, while for HLDA features which have been transformed discriminatively, the convergence was smoother and more stable, even improving at the 10-th iterations.

Table 1. Error rates (%) for MPE and fMPE for different features, on different acoustic levels

Error Rate (%)	MFCC			HLDA		
	Syllable	Character	Word	Syllable	Character	Word
Baseline (ML 40)	21.04	28.53	37.33	18.17	25.52	34.48
MPE (10 iterations)	17.53	24.89	33.43	15.43	22.71	31.19
fMPE (3 iterations)	16.56	24.01	32.74	14.28	22.09	30.92

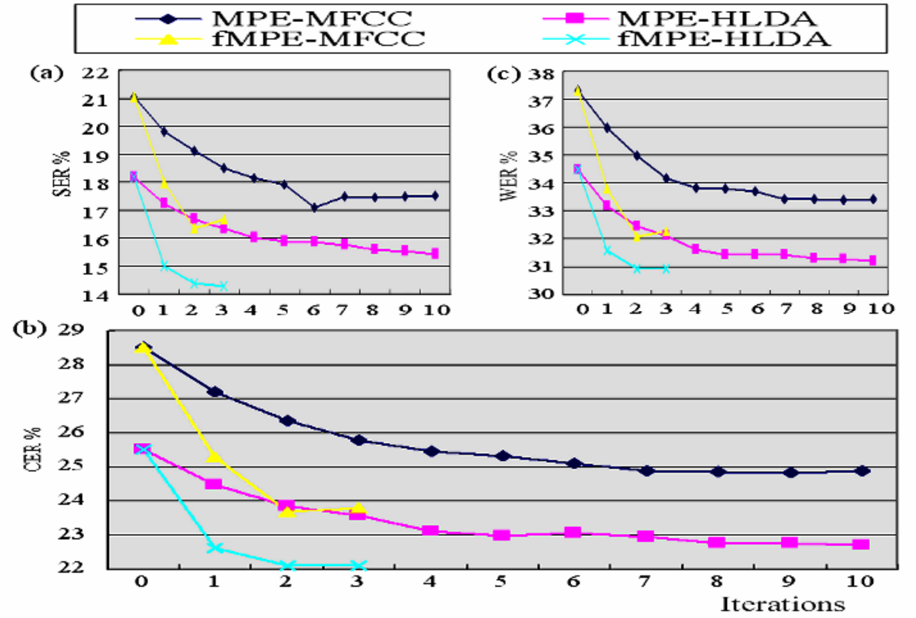


Fig. 2. Comparison of MPE and fMPE results for MFCC and HLDA features: (a) syllable error rates (SER), (b) character error rates (CER), and (c) word error rates (WER) as functions of numbers of iterations

6.3 fMPE Feature Training

14562 Gaussians and 9-context expansion were used in the baseline fMPE. As shown in the last row of Table 1, fMPE (3 iterations) significantly improved the performance on syllable, character and word levels regardless of the features used. For MFCC features the absolute error rate reduction was 4.48% in syllable error rate (SER) (21.04% to 16.56%), 4.52% in character error rate (CER) (28.53% to 24.01%), and 4.59% in word error rate (WER) (37.33% to 32.74%). Similar results were obtained for HLDA features, for example with an absolute CER reduction of 3.43% (25.52% to 22.09%) or 13.44% relative CER reduction. Note that both HLDA and fMPE transformed the features discriminatively, but fMPE offered extra improvements in addition to HLDA.

As shown in Fig. 2(a), (b) and (c), the performance of fMPE converges very quickly (2-3 iterations) for both MFCC and HLDA features as compared to MPE. From Fig. 2, fMPE gave more significant error reductions in the first iteration than in the second or the third. By examining the percentage of elements in the transform matrix which changed their signs (between positive and negative) after each iteration, we also found that this percentage decreased in each iteration. For HLDA features as an example, this percentage of elements which changed their signs are 39.86, 32.80, and 23.56 respectively for the first, second, and third iterations. For fMPE, the convergence was also faster, smoother and more stable for HLDA than for MFCC features. This seemed to indicate that more discriminative features are more robust to the iteration numbers or the learning rate of fMPE.

MPE makes the models closer to the features of the numerator lattice and farther away from those of the denominator lattice, while fMPE makes the features closer to the correct models and farther away from other confusing models. By estimating the entropy roughly by the posterior probabilities of the high-dimensional features, we also found that this entropy decreased in each iteration, which indicated the reduction of the degree of confusion.

6.4 Offset fMPE and Dimension-Weighted Offset fMPE

1908 Gaussians and 1-context expansion were used for both offset fMPE and dimension-weighted offset fMPE. The results of CER for both MFCC and HLDA features are shown in Fig. 3. Compared with Fig. 2(b), it can be found that here offset fMPE offered slightly higher CER than the original fMPE but using much smaller number of Gaussians. This is different from the results reported easier [4], where better performance can be obtained by offset fMPE with a longer context expansion and very delicate design of the iterations. However, the offset fMPE here still improved the performance significantly from baseline for both MFCC (CER 28.53% to 25.47%) and HLDA features (CER 25.52% to 23.61%). Similar improvements were also obtained on syllable and word levels.

Also shown in Fig. 3 are the results for the dimension-weighted offset fMPE proposed in this paper, which actually further reduced the CER to 25.23% for MFCC and to 23.44% for HLDA features at the second iteration. In addition, the dimension-weighted offset fMPE seemed to be more robust with respect to the choice of the learning rate parameter than the original offset fMPE. As shown in Fig. 4, if we change the parameter E (the inverse of learning rate) from E=5 to E=1, the error rate of offset fMPE increases significantly while the proposed dimension-weighted offset fMPE performed very stably here. This is probably because the dimension-weighted offset fMPE gives more flexibility for each dimension to determine the proper value of the offset by considering the posterior probability on each dimension, while the original offset fMPE only considers the posterior probability for a whole feature vector. In other words, some dimensions of the feature vector may dominate the offset for all the dimensions in the original offset fMPE, and the resulting features may even deteriorate due to the improper offset when the learning rate is set too large.

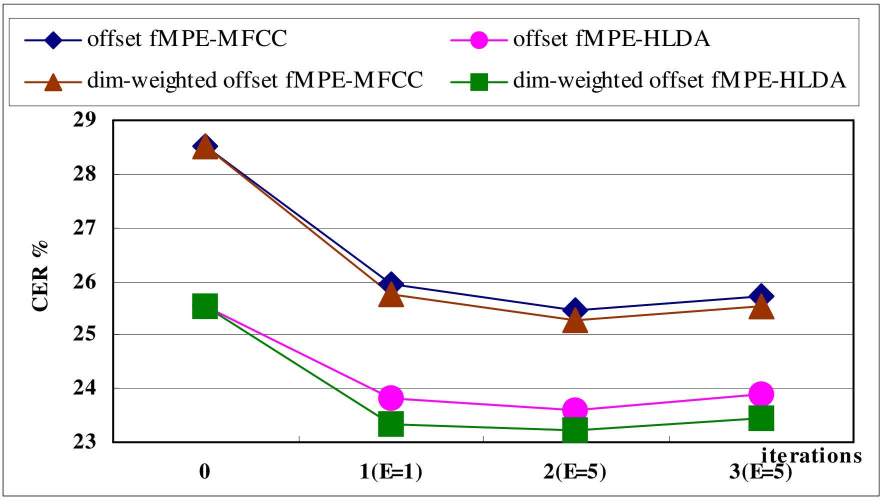


Fig. 3. CER(%) for offset fMPE and dimension-weighted offset fMPE with different features

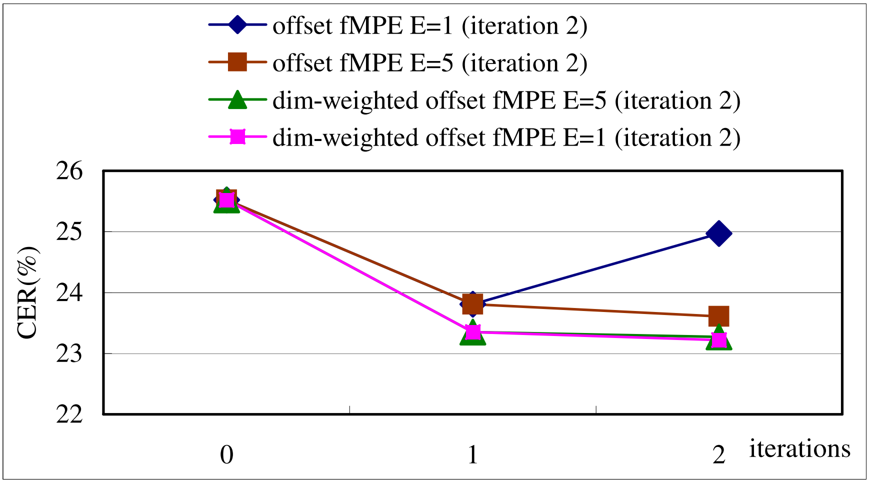


Fig. 4. Comparison of offset fMPE and dimension-weighted offset fMPE for different values of learning rate parameter E with HLDA features

7 Conclusion

This work reports the experiments and results in performing MPE, fMPE and offset fMPE on Mandarin Broadcast News task, and significant improvements in syllable, character and word error rate reductions were obtained. Trained from the same set of baseline models with HLDA features, MPE, fMPE (with 9- context expansion) and offset fMPE (with 1-context expansion) offered 2.81%, 3.43% and 1.91% character error rate reductions, respectively. In addition, the proposed dimension-weighted offset fMPE, which weights the offset on each dimension of a high-dimensional feature by the posterior probabilities of that dimension, is shown to be more robust with better performance than the offset fMPE. The integration of discriminative training and robust speech recognition techniques such as MMI-SPLICE [10] indicated promising extension and applications of the work in the future.

References

1. D. Povey, P. C. Woodland, "Minimum Phone Error and I-smoothing for Improved Discriminative Training," in Proc. ICASSP 2002.
2. D. Povey, "Discriminative Training for Large Vocabulary Speech Recognition," Ph.D Dissertation, Peterhouse, University of Cambridge, 2004.
3. D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, G. Zweig, "fMPE: Discriminatively Trained Features for Speech Recognition," in Proc. ICASSP 2005.
4. D. Povey, “Improvements to fMPE for Discriminative Training of Features,” in Proc. Interspeech 2005.
5. J. Kaiser, B. Horvat, Z. Kacic, "A Novel Loss Function for the Overall Risk Criterion Based Discriminative Training of HMM Models," in Proc. ICSLP 2000.
6. N. Kumar, "Investigation of Silicon Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition," PhD Dissertation, Johns Hopkins University, 1997.
7. B. Zhang, S. Matsoukas, "Minimum Phoneme Error Based Heteroscedastic Linear Discriminant Analysis for Speech Recognition," in Proc, ICASSP 2005.
8. R. A. Gopinath, “Maximum Likelihood Modeling with Gaussian Distributions for Classification,” in Proc. ICASSP 1998.
9. H.-M. Wang, B. Chen, J.-W. Kuo, S.-S Cheng, "MATBN: A Mandarin Chinese Broadcast News Corpus," International Journal of Computational Linguistics and Chinese Language Processing, 2005.
10. J. Droppo, A. Acero, "Maximum Mutual Information SPLICE Transform for Seen and Unseen Conditions," in Proc. Interspeech 2005.

State-Dependent Phoneme-Based Model Merging for Dialectal Chinese Speech Recognition

Linquan Liu, Thomas Fang Zheng, and Wenhu Wu

Center for Speech Technology, Tsinghua National Laboratory for
Information Science and Technology, Tsinghua University, Beijing, 100084
liulq@cst.cs.tsinghua.edu.cn, fzheng@tsinghua.edu.cn,
wuwh@tsinghua.edu.cn

Abstract. Aiming at building a dialectal Chinese speech recognizer from a standard Chinese speech recognizer with a small amount of dialectal Chinese speech, a novel, simple but effective acoustic modeling method, named *state-dependent phoneme-based model merging* (SDPBMM) method, is proposed and evaluated, where a tied-state of standard triphone(s) will be merged with a state of the dialectal monophone that is identical with the central phoneme in the triphone(s). It can be seen that the proposed method has a good performance however it will introduce a Gaussian mixtures expansion problem. To deal with it, an acoustic model distance measure, named *pseudo-divergence based distance measure*, is proposed based on the difference measurement of Gaussian mixture models and then implemented to downsize the model size almost without causing any performance degradation for dialectal speech. With a small amount of only 40-minute Shanghai-dialectal Chinese speech, the proposed SDPBMM achieves a significant absolute syllable error rate (SER) reduction of 5.9% for dialectal Chinese and almost no performance degradation for standard Chinese. In combination with a certain existing adaptation method, another absolute SER reduction of 1.9% can be further achieved.

Keywords: Speech recognition, dialectal Chinese speech recognition, state-dependent phoneme-based model merging, acoustic modeling, acoustic model distance measure.

1 Introduction

With regard to accented and dialectal speech recognition, a great deal of work has been done at various levels. Most of the dialect-specific automatic speech recognition (ASR) systems are concentrated on lexicon adaptation by capturing the pronunciation variations between standard speech and dialectal speech, and furthermore, characterizing these variation trends via a pronunciation lexicon [1-3]. Different from phone-level pronunciation modeling, the state-level pronunciation modeling is implemented to cover both the dialectal and the standard pronunciation characteristics [4, 5]. With regard to acoustic modeling, the adaptation techniques are most widely used through which dramatically significant improvement can usually be achieved [6, 7]. Some retraining mechanisms have also been proposed in which standard speech and dialectal/accented speech are pooled together [8]. Some researchers pay

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 282 – 293, 2006.

attention to language adaptation for accented speakers [9]. Additionally, the decoder is adjusted to cope with the differences between standard speech and dialectal speech [2]. In practice, these approaches are always integrated together to achieve much better performance in dialectal/accented speech recognition.

As far as acoustic modeling for accented speech recognition is concerned, a couple of methods are usually used, including: 1) *Adaptation*. The acoustic models trained with standard speech are transformed into accent-specific ones with a certain amount of accented speech by means of adaptation. The adaptation method has been applied by many researchers to the accented speech recognition with good results. However, while the pronunciations in the target accent/dialect being primarily considered, those in the original accent/dialect cannot be sufficiently covered simultaneously at acoustic level. 2) *Retraining*. It is the most straightforward approach that pools accented training data with standard data so as to retrain the acoustic models using combined data. In [10], it is shown that by simply pooling 34 hours of standard data with 52 minutes of accented data the word error rate can be reduced from 49.3% to 42.7%. Although significant improvement was achieved with a small amount of accented data for "*pooled*" training, an obvious disadvantage was that the retraining was dramatically time-consuming. 3) *Combination of acoustic modeling with state-level pronunciation modeling* [4, 5]. In [4], state-level pronunciation modeling was integrated with acoustic modeling to better characterize the phone changes in which a syllable error rate (SER) reduction of approximately 2.39% was achieved for spontaneous speech recognition. The problem is that a large amount of accented speech data is needed and that the proposed method is sometimes too complicated to be readily applied. 4) *Dialect detection* [11, 12]. It is often used as a front-end in state-of-the-art ASR systems. In this method, dialect-specific recognizers have to be built for each dialect or sub-dialect, which also needs a large amount of dialectal data, and the performance, relies heavily on the outcome of dialect detection.

In China, *Putonghua* (or standard Chinese) is the official language through which people from different regions can be mutually understood. In addition to *Putonghua*, there are other 8 major dialects, which can be detailedly divided into over 40 sub-dialects [6] or over 1,000 sub-sub-dialects [13]. *Putonghua* spoken by most Chinese people is usually influenced by their native dialect more or less. In this paper, we refer to *Putonghua* influenced by a certain Chinese dialect as *dialectal Chinese*. One of our motivations here is to build a robust recognizer for a certain dialectal Chinese based on the handy *Putonghua* model with a small amount of dialectal speech data (less than one hour). To build a robust and practical dialectal Chinese-specific recognizer, the following four requirements should be met: 1) the modeling method as simple as possible, which is a prerequisite for fast deployment of ASR systems; 2) only a small amount of dialectal speech data needed. In China, there are so many dialects that it is impossible to collect a large amount of speech data for each dialectal Chinese due to some economical considerations; 3) good performance in dialectal speech recognition as well as standard speech recognition. Essentially, a dialect-specific recognizer is regarded as the extension of a *Putonghua* recognizer. It is natural that the better performance should be obtained for dialectal Chinese speech recognition without (or almost without) any performance degradation for *Putonghua* speech recognition; 4) a complementary or additive approach to the existing adaptation techniques. It is generally believed that adaptation is one of the most effective ways for speech recognition of a dialectal Chinese of interest. Hopefully, the proposed modeling method can be used as a complement for the adaptation techniques in order to further improve the performance.

In order to reach the goal mentioned above, a novel, simple but effective acoustic modeling method is proposed in this paper, named as *state-dependent phoneme-based model merging* (SDPBMM) method. In SDPBMM, based on a same phoneme, the state-level parameters from a context-dependent *Putonghua* HMM and its phoneme-related context-independent dialectal HMM are merged according to a certain criterion. The idea comes from the assumption that the HMM from standard speech can "*borrow*" some information from its corresponding HMM in the target dialectal speech in order to reduce the differences between the dialectal speech and the standard speech. To a great extent, the newly-merged HMM can cover both dialectal and standard speech acoustically. In this paper, with only 40-minute Shanghai-dialectal speech data adopted, a cost-effective acoustic model for the target dialectal Chinese can be built from the *Putonghua* recognizer using SDPBMM method. It is experimentally shown that SDPBMM is able to meet the foresaid four requirements.

As a side effect of SDPBMM, the number of Gaussian mixtures within the merged HMMs is increased definitely, we regard it as a Gaussian mixtures expansion problem. To deal with it, an acoustic distance measure, named *pseudo-divergence based distance measure* (PDBDM), is proposed based on the difference measurement of Gaussian mixture models, and then implemented under the assumption that the similarity between two states can be measured by an acoustic distance between them. As a result, PDBDM can differentiate the states that need model merging from those that do not need merging in SDPBMM. More importantly, PDBDM can downsize the parameter scale of HMMs almost without causing any performance degradation on dialectal Chinese speech.

The remainder of this paper is organized as follows. The basic ideas of the SDPBMM will be described comprehensively in Section 2. In Section 3, a merging criterion, namely PDBDM, will be introduced which is to reduce the parameter scale in the SDPBMM-based HMMs. A series of experiments designed to evaluate the effectiveness of the proposed methods as well as the experimental results will be presented in Section 4. Finally, conclusions are drawn and future work is suggested in Section 5.

2 State-Dependent Phoneme-Based Model Merging

2.1 Description and Formulation of SDPBMM

In [4], a state-level pronunciation modeling method, the *partial change phone models,* was proposed, which could cover both the base form pronunciation and the surface form pronunciation simultaneously. The actually realized pronunciations except for the canonical pronunciation were merged with the pre-trained base form-based acoustic models in terms of the *acoustic model reconstruction*. Inspired by the idea, we make an attempt to take the standard pronunciation and dialectal pronunciation into consideration in acoustic modeling. In SDPBMM, the context-dependent HMMs for standard Chinese are merged with their phoneme-related context-independent HMMs for dialectal Chinese at state level. In other words, the "*correct*" (base form) pronunciations in dialectal speech are involved in the merging instead of "*wrong*" (surface form) pronunciations adopted in [4, 5]. Due to the data sparseness issue, only monophone HMMs for dialectal Chinese are considered in SDPBMM. Compared with the *acoustic model reconstruction* based on triphone HMMs, a remarkably small amount of dialectal data is needed to build monophone HMMs and no further training process is necessary.

Most the state-of-the-art ASR systems tend to use context-dependent triphone HMMs to achieve a higher accuracy. In order to reduce the model complexity, downsize the redundant Gaussian components and re-estimate the unseen triphones in training data, the decision tree based state tying method is commonly used [14]. The states from some triphones with the same central phoneme are presented by a decision tree in which the tied states are presented by a leaf node. The idea is illustrated in Figure 1. In the left part of Figure 1, all the second states of the *an*-centered triphones are presented by a decision tree. In addition, both a state of monophone from dialectal speech and a state of triphone from standard speech are composed of multiple Gaussian mixtures. To accomplish the merging, the second state from the dialectal monophone *an* is merged with the leaf nodes of *an*-centered decision tree, *i.e.* the tied states. The merging process is depicted in the right part of Figure 1. The merging takes place between a monophone from dialectal speech and a triphone from standard speech whose central phoneme is the same as the monophone at the state level. As a result, a merged tied-state consists of multiple Gaussian mixtures from both the state of standard triphone HMM and its corresponding state of dialectal monophone HMM, as denoted by black solid curves and red dotted curves in Figure 1, respectively.

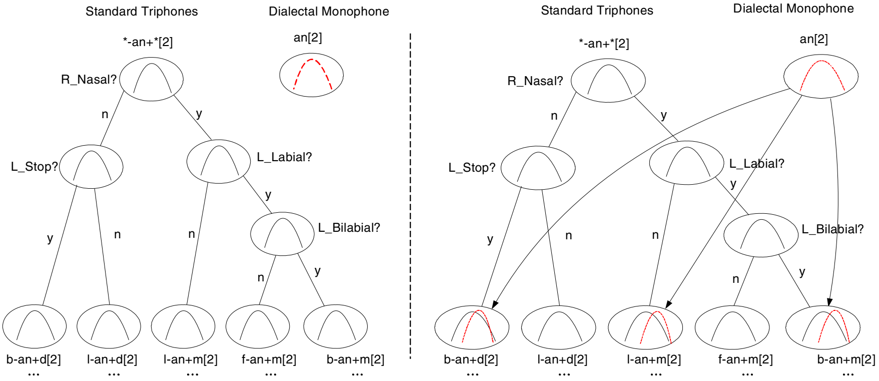


Fig. 1. The topology before and after the application of SDPBMM

Let x, s, and d be an input vector, a state from standard speech, and a state from dialectal speech, respectively, the original probability density function for continuous density HMM $P(x|s)$ is

$$P(x|s) = \sum_{k=1}^{K} w_{sk} N\left(x; \mu_{sk}; \Sigma_{sk}\right) \ , \tag{1}$$

where w_{sk} is the mixture weight of k-th mixture component of state s, K is the total number of Gaussian mixtures in state s. For simplification, $N_{sk}(\cdot)$ will be used to denote $N(x;\mu_{sk};\Sigma_{sk})$ of state s hereinafter.

Let $P'(x|s)$ be the revised output distribution of a merged state after applying SDPBMM, it can be represented as

$$P'(x|s) = \lambda P(x|s) + (1-\lambda)P(x|s,d)P(d|s) \ , \tag{2}$$

where λ is a linear interpolating coefficient between the standard and the dialectal acoustic models and is usually determined experimentally, and $P(d|s)$ can be regarded as a kind of pronunciation modeling. Because the purpose here is to verify the effectiveness of SDPBMM, the pronunciation variations between standard pronunciation and dialectal pronunciation are not taken into consideration in this paper and therefore we set $P(d|s) \equiv 1$. Afterwards, Equation 2 can be further simplified as Equations 3 and 4.

$$P'(x|s) = \lambda P(x|s) + (1-\lambda)P(x|d) \ , \tag{3}$$

$$\begin{aligned} P'(x|s) &= \lambda\sum_{k=1}^{K} w_{sk} N_{sk}(\cdot) + (1-\lambda)\sum_{n=1}^{N} w_{dn} N_{dn}(\cdot) \\ &= \sum_{k=1}^{K} \lambda w_{sk} N_{sk}(\cdot) + \sum_{n=1}^{N} (1-\lambda) w_{dn} N_{dn}(\cdot) \\ &= \sum_{k=1}^{K} w'_{sk} N_{sk}(\cdot) + \sum_{n=1}^{N} w'_{dn} N_{dn}(\cdot) \ . \end{aligned} \tag{4}$$

Equation 3 is actually a kind of interpolation method [8]. In Equation 4, K and N are the numbers of Gaussian mixtures of state s from standard speech and state d from dialectal speech, respectively; nevertheless parameters K and N are not necessary to be equal to each other. Parameters w'_{sk} and w'_{dn} are new mixture weights in a merged state of SDPBMM, just as indicated in Equation 4, $w'_{sk} = \lambda w_{sk}$ and $w'_{dn} = (1-\lambda)w_{dn}$.

2.2 Analysis

In SDPBMM, it is very easy to build context-independent monophone HMMs via just a quite small amount of dialectal data, and the merging is performed based on a standard triphone decision tree at state level. The SDPBMM-based acoustic model does not need retraining, which will save time and efforts dramatically. In essence, the SDPBMM-based acoustic model is still a standard recognizer just with much more acoustic coverage on dialectal speech, and so it is expected to be able to achieve good performance for both dialectal speech and standard speech recognition.

3 Pseudo-divergence Based Distance Measure

With the application of SDPBMM, the acoustic coverage is enlarged so that the accuracy for dialectal speech recognition can be improved; however, the Gaussian mixtures in the merged states are definitely increased. The efficiency is lowered due to much time consumption during the decoding procedure. For example, when a standard state consisting of 14 Gaussians is merged with a dialectal state of 6 Gaussians, the number of Gaussian mixtures is increased by 43% and thereby the time consumption is increased by 56% if all standard states are involved in the merging process. This is a Gaussian mixtures expansion problem. To deal with it, a mechanism has to be proposed to tell the states that need merging from those that do not need merging. Intuitively, there exists a different level similarity among states of dialectal monophone and states of standard triphone. Presumably, some measures can be taken to evaluate the similarity which can act as a criterion to classify the states participating in merging process. In practice, the similarity can be measured by the distance between two states instead.

In HMMs, each state is represented by a probability distribution function (PDF) in terms of mixed Gaussian mixtures. Several approaches have been proposed to measure the distance between two HHM states. 1) The relative entropy or Kullback Leibler distance (KLD) [15], which can represent the distance comprehensively but accordingly the computation complexity will easily go beyond control with the increased dimension. 2) Extended KLD, which is a practical way to approximate the distance [16]. But it can not be used to deal well with mixed-mixture PDFs and great time consumption is required. 3) Parametric distance metric for mixture PDF [17], which can effectively measure the distance directly between PDFs with mixed mixtures from the model's parameter. Actually, this approach is an issue of linear programming and can be solved via simplex tableau. However, sometimes the optimal solution can not be obtained under some rigid constraints. In this paper, as a tradeoff between precision and efficiency, a distance measure, named *pseudo-divergence based distance measure* (PDBDM), which was initially used and implemented in speaker recognition [18], is modified here to act as the distance measure between a state of dialectal monophone HMM and a tied-state of standard triphone HMM.

3.1 Basic Idea of PDBDM

In this section, the basic idea of PDBDM is to be illustrated in detail. First, the *dispersion* between two HMM states is defined as

$$dispersion(A,B)=\sum_{i=1}^{M} w_{Ai}\left[\sum_{j=1}^{N} w_{Bj} d_{A,B}\left(i,j\right)\right], \tag{5}$$

where A and B are two HMM states, $d_{A,B}(i, j)$ is the distance between the i-th mixture from A and j-th mixture from B. M and N are the total numbers of Gaussian mixtures in A and B, respectively. Accordingly, the *self-dispersion* is

$$dispersion(A,A)=\sum_{i=1}^{M} w_{Ai}\left[\sum_{j=1}^{M} w_{Aj} d_{A,A}\left(i,j\right)\right]. \tag{6}$$

Then the *pseudo-divergence* between two HMM states is formulated as[1]:

$$pseudo\text{-}divergence\left(\lambda_A,\lambda_B\right)=\frac{dispersion(A,B)}{dispersion(A,A)}\ , \tag{7}$$

$$pseudo\text{-}divergence\left(\lambda_B,\lambda_A\right)=\frac{dispersion(B,A)}{dispersion(B,B)}\ . \tag{8}$$

Usually $pseudo\text{-}divergence\left(\lambda_A,\lambda_B\right)\neq pseudo\text{-}divergence\left(\lambda_B,\lambda_A\right)$. To minimize the statistical difference, the distance between two HMM states is redefined as

$$distance\left(\lambda_A,\lambda_B\right)=\frac{1}{2}\left(\frac{dispersion(A,B)}{dispersion(A,A)}+\frac{dispersion(B,A)}{dispersion(B,B)}\right)\ . \tag{9}$$

As for the distance between two single Gaussian mixtures, *i.e.* $d_{A,B}(i,j)$ in Equation 5, there are normally four options, the *Euclidean* distance measure, the *Mahalanobis* distance measure, the *weighted Mahalanobis* distance measure and the *Bhattachyaryya* distance measure. The Bhattachyaryya distance measure is adopted here because it is thought to be able to characterize the distance more precisely by taking the difference of covariance into account [17]. Given two Gaussian mixtures, $\lambda_1(\mu_1, \Sigma_1)$ and $\lambda_2(\mu_2, \Sigma_2)$, the Bhattachyaryya distance measure is defined as

$$d(\lambda_1,\lambda_2)=\frac{1}{8}(\mu_1-\mu_2)^T\left(\frac{\Sigma_1+\Sigma_2}{2}\right)^{-1}(\mu_1-\mu_2)+\frac{1}{2}\ln\frac{\left|{(\Sigma_1+\Sigma_2)}/{2}\right|}{\left|\Sigma_1\right|^{1/2}\left|\Sigma_2\right|^{1/2}}\ . \tag{10}$$

3.2 Combination of SDPBMM with PDBDM

A state from a dialectal monophone HMM and its corresponding state on a basis of the same phoneme from a standard triphone HMM form a pair for the calculation of distance. The distances of all pairs are computed using Equation 9. Subsequently, a certain percentage, *i.e.* 70% relative to the amount of pairs, is set as a *threshold* in the descending order of distance so that the pairs with a large distance have a higher priority to be chosen to participate in the merging. The idea is depicted in Equation 11.

$$\begin{cases} merging, & \text{distance}(d,s)\geq threshold \\ no\text{-}merging, & \text{distance}(d,s) < threshold \end{cases} \tag{11}$$

The application of PDBDM in SDPBMM is based on the assumption that the distance can be used to characterize the similarity between two states instead, but in a reverse sense that a smaller distance corresponds to a bigger similarity. If the distance between two states is small, it can be safely inferred that there is less variability between them and in which case no merging is necessary because the original state

[1] In the paper, the concept of *divergence* is not completely same as the classic definition of *divergence*, so *pseudo-divergence* is named.

from the standard speech has already covered the acoustic space sufficiently. As for the pairs with big distances, the merging is performed to cover both the standard and the dialectal speech acoustically. Notice the fact that in the right part of Figure 1, some states, for example *l-an+d*[2] and *f-an+m*[2], are not involved in the merging, which represents the purpose of PDBDM. It is expected that the scale of Gaussian mixtures can be downsized by PDBDM while no performance degradation takes place for dialectal speech recognition.

4 Experiments and Results

The Mandarin Broadcast News (MBN) database (Hub4NE), a read style standard Chinese speech corpus, was used to train the baseline system, the *Putonghua* recognizer. It contained about 30 hours of high quality wideband speech with detailed Chinese Iinital/Final (IF) transcriptions. The acoustic models of *Putonghua*-based baseline were tied-state cross-word standard tri-IF HMMs. Each tri-IF was modeled using a left-to-right non-skip 3-state continuous HMM, with 14 Gaussian mixtures per state. 39-dimensional MFCC coefficients with Δ and ΔΔ were used as features with cepstral mean normalization [19]. The HMMs achieve good performance statistically upon which many research was carried out [12]. Additionally, 6 zero-Initials were added to the standard IF set to help improve the performance and make the modeling process consistent. Another database, namely Wu dialectal Chinese database (WDC) [20], contained 100 native Shanghai speakers, 50 males and 50 females. The speech data of WDC was recorded under a similar condition to that of MBN. The use of this database was to minimize the channel affect. The WDC was composed of the speech from medium and strong Shanghai-accented speakers. Further details on the database can be found in [20]. Adopted in the following experiments as the recognition lexicon were 406 toneless Chinese syllables.

Three data sets were selected from the WDC and MBN, one was the development training set, *Dev_WDC*, which consisted of about 40-minute Shanghai-dialectal Chinese speech by 10 speakers. The *Dev_WDC* was used to build 65 context-independent dialectal mono-IF HMMs for SDPBMM, each monophone HMMs was of the exactly same topology as that of standard tri-IFs except that there were 6 Gaussian mixtures per state. Another data set was *Test_WDC* composed of 20 speakers' speech from the WDC. The third data set was *Test_MBN* from MBN also used for testing. The three data sets were not overlapping with one another. The detailed information for the data sets used in the experiments is listed in Table 1. Initially, the MBN-based *Putonghua* HMMs achieved SERs of 30.5% and 49.8% on *Test_MBN* and *Test_WDC*, respectively; there was an absolute degradation of approximately 20% on the Shanghai-dialectal Chinese speech. An SER of 54.1% on *Test_WDC* was achieved by the dialectal mono-IF HMMs built upon *Dev_WDC*. Because acoustic modeling was our research focus no language models were used. Our experiments were performed at the syllable level and the SER reduction was used as a measure of the improvement. Besides, HTK [21] was used in the experiments.

Table 1. Detailed information for the development and test sets

Data set	Database	Details
Dev_WDC	WDC	10 speakers, 510 utterances, totally 40-minute speech
Test_WDC	WDC	20 speakers, 995 utterances, totally 60-mintue speech
Test_MBN	MBN	1,200 utterances, totally 80-minute speech

The linear coefficient in Equation 3 was determined experimentally and λ was set to 0.72. With the application of PDBDM, there were 70% of tied states from *Putonghua* tri-IFs involved in the SDPBMM. The recognition results on the dialectal test set, *Test_WDC*, are listed in Table 2. It can be seen that the SDPBMM can reduce the SER by 6.2% absolutely on dialectal speech with only 40-minute dialectal Chinese speech data. However the number of Gaussian mixtures was increased by approximately 43%. To deal specifically with the Gaussian mixtures expansion problem, the PDBDM was adopted with the expectation that no degradation is introduced. Thus, the number of Gaussian mixtures in *SDPBMM+PDBDM* was decreased by 30% with a slight SER increase of 0.3% absolutely. Compared with the baseline, an absolute SER reduction of 5.9% was still achieved by the *SDPBMM+PDBDM*. It is shown that PDBDM can downsize the parameter scale without significant performance degradation. In the following experiments, *SDPBMM+PDBDM* was used as the default SDPBMM-based acoustic modeling.

Table 2. The results for *Putonghua*, SDPBMM, and SDPBMM in conjunction with PDBDM on *Test_WDC*

	Putonghua	SDPBMM	SDPBMM+PDBDM
States	3,230	3,230	3,230
Gaussians	45,220	64,600	58,786
Tri-IFs	7,411	7,411	7,411
SER	49.8%	43.6%	43.9%

4.1 Comparison Conditioned on Same Amount of Gaussian Mixtures

As for SDPBMM-based acoustic model, it is naturally assumed that the improvement in dialectal speech recognition may result from the increase of Gaussian mixtures in the merged states. Compared with the *Putonghua* HMMs with 14 Gaussian mixtures per state, on average, there were 18.2 mixtures per state in SDPBMM-based HMMs. To make a fair comparison, another *Putonghua* acoustic model with 18 Gaussian mixtures per state was generated which had approximately equal parameter scale as that of the SDPBMM. The SER on *Test_WDC* was decreased from 49.8% to 49.1% compared with the baseline, but there still existed an SER gap of 5.2% absolutely in comparison with the SDPBMM. It is shown that increasing the parameter scale solely can not achieve significant improvement in dialectal speech recognition.

4.2 Evaluation on Standard Speech Recognition

The effectiveness of SDPBMM-based acoustic model on standard speech recognition can be seen from the results listed in Table 3 with *Test_MBN* taken as the test set. It is shown that as expected, the SDPBMM can achieve a slightly higher SER (an absolute 0.6% higher) on standard speech than the *Putonghua* acoustic model. It could be concluded that the SDPBMM can achieve significant improvement in dialectal speech recognition without significant degradation in standard speech recognition.

Table 3. The results for *Putonghua* and SDPBMM on *Test_MBN*

	Putonghua	SDPBMM
SER	30.5%	31.1%

4.3 Integration with Adaptation

Adaptation is one of the most effective ways for dialectal speech recognition. Most widely used adaptation techniques include the maximum linear likelihood regression (MLLR) and the maximum *a posteriori* adaptation (MAP) methods [19]. For comparison, the adaptation was performed with exactly the same amount of dialectal speech data as in the experiment regarding SDPBMM. Considering that MLLR is much beneficial when there is only a small amount of adaptation data available [7], we adopted MLLR for model adaptation. The MLLR adaptation was performed based on the *Putonghua* acoustic model, denoted as *MLLR*, in which all the standard tri-IFs were classified into 65 classes, and mean update was performed in transformation matrix. Note that, *Dev_WDC* was also used as the adaptation data in MLLR adaptation. As a result, an SER of 44.1% was achieved on *Test_WDC* which was still slightly higher than the SER of 43.9% by SDPBMM with exactly the same data set. The results are listed in columns *SDPBMM* and *MLLR* in Figure 2, respectively. It is shown that compared with MLLR, SDPBMM can achieve a comparable performance on dialectal speech recognition with only a small amount of dialectal data available.

In addition, it is assumed that SDPBMM primarily concentrates on addressing the issues of the phonetic mismatch between the dialectal speech and the standard speech. As a matter of fact, the adaptation can be a good solution to channel mismatch. Therefore it is expected that the SDPBMM in combination with a certain adaptation method can have the potential to further improve the performance on dialectal speech recognition. To verify the assumption, another development data set of Shanghai-dialectal Chinese, *Dev_WDC*1, was selected from WDC database, which consisted of 410 utterances by 10 speakers (approximately 30 minutes). By using *Dev_WDC* and *Dev_WDC*1, two new acoustic models were built, namely *SDPBMM+MLLR* and *MLLR+SDPBMM*, where the order in the names means the order that the components were performed. In *SDPBMM+MLLR*, the SDPBMM was performed using *Dev_WDC* based on *Putonghua* HMMs followed by the MLLR adaptation using *Dev_WDC*1; and vice versa. The results are also listed in Figure 2. From the figure, it can be clearly seen that in combination with the MLLR adaptation, another two absolute SER reductions of 1.9% and 1.8% on dialectal Chinese speech can be further achieved by *SDPBMM+MLLR* and *MLLR+SDPBMM*, respectively. The results correspond to columns *SDPBMM+MLLR* and *MLLR+SDPBMM* in Figure 2, respectively. Another phenomenon is that *SDPBMM+MLLR* and *MLLR+SDPBMM* achieved approximately an equal SER, which is to say, the SDPBMM and MLLR can collaborate perfectly irregardless of the application order. In conclusion, SDPBMM and MLLR are additive and exchangeable algebraically.

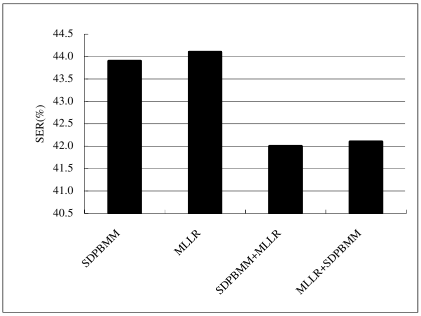


Fig. 2. Comparison with MLLR adaptation on *Test_WDC* and integration with MLLR adaptation on *Test_WDC* and *Test_WDC*1

5 Conclusions and Future Work

In the paper, a novel, simple but effective acoustic modeling method for dialectal Chinese speech recognition, SDPBMM, is proposed. Though it will introduce a Gaussian mixtures expansion problem, a corresponding PDBDM acting as a merging criterion is proposed to be integrated into SDPBMM, which can result in no significant degradation for dialectal Chinese speech recognition. From a series of experiments, it can be concluded that the SDPBMM has the advantages: 1) It is simple but practical for acoustic modeling when there is quite a small amount dialectal speech data available; 2) It can make a significant performance improvement for dialectal speech recognition; 3) It can have good performance for both standard and dialectal speech recognition; 4) It can achieve comparable performance to adaptation with only a small amount dialectal speech data available; 5) It is additive to adaptation, that is to say, the application of SDPBMM and adaptation in any order can further improve the performance for dialectal speech recognition. In a word, the SDPBMM is one of the most effective acoustic modeling methods for read-style dialectal Chinese speech recognition. In this paper, the experiments were done on Shanghai-dialectal Chinese, but no dialect-specific prior knowledge is incorporated in SDPBMM, thus, this method can be easily generalized to other dialectal Chinese.

Another issue is that the experiments in this paper were based on read speech. In our next step the research on spontaneous speech will be carried out where pronunciation modeling [22] should be taken into account. It is believed that the use of pronunciation modeling can help build much precise acoustic model to better characterize pronunciation variations between dialectal Chinese and *Putonghua*, not only for spontaneous speech but also for read speech.

References

1. Goronzy, S., Kompe, R., Rapp, S.: Generating Non-Native Pronunciation Variants for Lexicon Adaptation. Speech Communication, Vol. 42(1):109-123, 2004
2. Huang, C., Chen, T., Chang, E.: Accent Issue in Large Vocabulary Continuous Speech Recognition. International Journal of Speech Technology, 7: 141-153, 2004

3. Tjalve, M., Huckvale, M.: Pronunciation Variation Modeling using Accent Features. Proc. Interspeech, 2005, Lisbon
4. Liu, Y., Fung, P.: Pronunciation Modeling for Spontaneous Mandarin Speech Recognition. International Journal of Speech Technology, 7:155-172, 2004
5. Saraclar, M., Nock, H., Khudanpur, S.: Pronunciation Modeling by Sharing Gaussian Densities across Phonetic Models. Computer Speech and Language, 14:137–160, 2000
6. Li, J., Zheng, T.-F., Byrne, W., Jurafsky, D.: A Dialectal Chinese Speech Recognition Framework. Journal of Computer Science and Technology, 21(1): 106-115, Jan. 2006
7. Diakoloukas,V., Digalakis, V., Neumeyer, L., Kaja, J.: Development of Dialect-Specific Speech Recognizers Using Adaptation Methods. IEEE ICASSP, 2:1455, 1997.
8. Tomokiyo, L.-M.: Recognizing Non-Native Speech: Characterizing and Adapting to Non-Native Usage in LVCSR. PhD Thesis, Carnegie Mellon University, 2001.
9. Gao, J.-F., Goodman, J., Li, M.-J., Lee, K.-F.: Toward a Unified Approach to Statistical Language Modeling for Chinese. ACM Transactions on Asian Language Information Processing, 1(1): 3- 33, March 2002
10. Wang, Z.-R., Schultz, T., Waibel, A.: Comparison of Acoustic Model Adaptation Techniques on Non-native Speech. IEEE ICASSP, 540–543, 2003
11. Zheng,Y.-L., Sproat, R., Gu, L. *et al.*: Accent Detection and Speech Recognition for Shanghai-Accented Mandarin", Interspeech 2005, Lisbon
12. Sproat, R., Zheng, T.-F., Gu, L., Jurafsky, D., Shanfran, I., Li, J., Zheng, Y.-L., Zhou, H., Su, Y., Tsakalidis, S., Bramsen, P., Kirsch, D.: Dialectal Chinese Speech Recognition: Final Technical Report. 2004, http://www.clsp.jhu.edu/ws2004/
13. Li, A.-J., Wang, X.: A Contrastive Investigation of Standard Mandarin and Accented Mandarin, EuroSpeech, 2003, Geneva
14. Hwang, M.-Y., Huang, X.-D., Alleva, F.-A.: Predicting Unseen Triphones with Senones, IEEE Transaction on Speech and Audio Processing, 4(6):412-419, 1996
15. Cover, T.-M., Thomas, J.-A.: Elements of Information Theory, John Wiley & Sons, 1991
16. Liu, Z., Huang, Q.: A New Distance Measure for Probability Distribution Function of Mixture Types. Proc. ICASSP, 1345-1348, 2000
17. Liu, Y., Fung, P.: Acoustic and Phonetic Confusions in Accented Speech Recognition. Proc. INTERSPEECH, 3033-3036, 2005
18. Xuan, P., Wang, B.-X.: Speaker Clustering via Distance Measurement of Gaussian Mixtures Models. Journal of Computer Engineering and Technology, May, 2005
19. Huang, X.-D. Acero, A., Hon, S.-W.: Spoken Language Processing, Prentice Hall, 2001
20. Li, J., Zheng, F., Xiong, Z.-Y. Wu, W.-H.: Construction of Large-Scale Shanghai Putonghua Speech Corpus for Chinese Speech Recognition, Oriental-COCOSDA, 62-69, October, 2003, Singapore
21. Young, S., Evermann,G., Hain,T. *et al.*: The HTK Book (for HTK Version 3.2.1). Cambridge University, Cambridge, 2002. http://htk.eng.cam.ac.uk/
22. Zheng, F., Song, Z.-J., Fung, P., Byrne, W.: Mandarin Pronunciation Modeling Based on CASS Corpus, Journal of Computer Science and Technology, 17(3): 249-263, May 2002

Non-uniform Kernel Allocation Based Parsimonious HMM

Peng Liu, Jian-Lai Zhou, and Frank Soong

Microsoft Research Asia, Beijing, 100080
{pengliu, jlzhou, frankkps}@microsoft.com

Abstract. In conventional Gaussian mixture based Hidden Markov Model (HMM), all states are usually modeled with a uniform, fixed number of Gaussian kernels. In this paper, we propose to allocate kernels non-uniformly to construct a more parsimonious HMM. Different number of Gaussian kernels are allocated to states in a non-uniform and parsimonious way so as to optimize the Minimum Description Length (MDL) criterion, which is a combination of data likelihood and model complexity penalty. By using the likelihoods obtained in Baum-Welch training, we develop an effcient backward kernel pruning algorithm, and it is shown to be optimal under two mild assumptions. Two databases, Resource Management and Microsoft Mandarin Speech Toolbox, are used to test the proposed parsimonious modeling algorithm. The new parsimonious models improve the baseline word recognition error rate by 11.1% and 5.7%, relatively. Or at the same performance level, a 35-50% model compressions can be obtained.

1 Introduction

In the state-of-the-art of Automatic Speech Recognition (ASR), Hidden Markov Models (HMMs) have been successfully applied to modeling the dynamic evolution of speech signals, and Gaussian Mixture Models (GMMs), which is quite flexible to approximate various distributions, are the most popular statistical models in modeling the output observations of HMM states. In a typical ASR system, we just assign each phoneme an identical HMM topology, and model each state with a fixed number of Gaussian kernels.

However, given the well-known Occam's razor [1] "*Given two equally predictive theories, choose the simpler*," we should search for a parsimonious model for better performance and robustness. In general, the purpose of parsimony is to build statistical models with adequate topology and number of parameters, which can be considered from two viewpoints: 1) how to determine the number of parameters of a system? 2) given the total number of parameters, how to allocate parameters optimally to each part of the system? Typically, the two problems can be solved by exploiting model complexity penalization criterion, e.g. Akaike Information Criteria (AIC) [2], Bayesian Information Criteria (BIC) [3] and Minimum Description Length (MDL) [4].

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 294–302, 2006.

In this study, we apply MDL criterion to speech model training. By parsimonious modeling, we aim at constructing a recognition system with good classification performance and compressing models to achieve comparable performance but with significantly fewer parameters [5].

In HMM based ASR system, we can find parsimonious models with more compact model topologies, e.g. number of HMMs, number of states in each HMM, and/or the number of Gaussian kernels in each state. For the first issue, successive state spliting (SSS) [6] provides us a good generic solution. In this paper, we concentrate on the second issue. Namely, how to optimally allocate different numbers of kernels to each state in an HMM system. A couple of reasonable assumptions are adopted to simplify solution. By making use of the by-products in conventional mixture-up Baum-Welch training process, we develop an effective and efficient kernel allocation algorithm.

The rest of the paper is organized as follows: In Section 2, a general optimization framework of parsimonious Gaussian kernel assignment is presented; in Section 3, we simplify the optimization problem for an effective and efficient step back algorithm; in Section 4, experimental results are given and discussed; conclusions are finally given in section 5.

2 MDL Based Gaussian Kernal Allocation for HMMs

States in HMMs are usually modeled as a mixture of uniform, fixed number of Gaussian kernels [7]. However, conceptually, we tend to believe that each state output distribution should characterized by different number of Gaussian kernels. By allocating Gaussian kernels non-uniformly across different states, we can maximize the modeling efficiency. We give an example of a 6-component Gaussian mixture of C_1 in Figure 1. The distribution can be probably modeled well by a 2-component mixture without sacrificing and modeling resolution.

Maximum Likelihood (ML) criterion is widely adopted in acoustic modeling, and the Baum-Welch algorithm [8] has been developed to train HMMs. To prevent overtraining the model penalized by model complexity, ML can be modified to MDL. Here we make effort to assign appropriate number of Gaussian kernels to each state in the MDL sense based on the conventional training process. Denote J the total number of states, T the total number of frames in the training set, and d the feature demension, MDL criterion used in HMM training can be written as:

$$C(\ m) = \underbrace{\sum_{j=1}^{J} \sum_{t=1}^{T} \gamma_j(t) \log b_j(o_t)}_{L(\ m)} - \underbrace{\frac{K}{2} \log(Td)}_{P(\ m)} \tag{1}$$

where $m = (m_1, \ldots, m_J)^\top$ is the configuration of number of kernels, with denoting m_j the number of kernels of the j^{th} state $K = \sum_{j=1}^{J} m_j(1 + 2d) - J$ is the total number of parameters in the system; $\gamma_j(t)$ and $b_j(o_t)$ are the occupancy and likelihood of the j^{th} state, given the t^{th} training frame, respectively, both of them depents on m. The two terms $L(m)$ and $P(m)$ in this criterion are likelihood and panelty of the model complexity respectively. Actually, we have various choices for the penalty term, here the simplest form which is equivalent to BIC is adopted.

By adopting MDL criterion, parsimonious HMM modeling in terms of kernel allocation is given by the following optimization:

$$\hat{m} = \text{argmax}_{\ m} C(m) \qquad (\text{s.t.} \sum_{j=1}^{J} m_j = J M_{\text{T}}) \tag{2}$$

where M_{T} is the target average number of Gaussian kernels per state. Here the constraint in number of kernels is optional: Without the constraint, we are finding the global optimal configuration in MDL sense. With the constraint, we are finding the optimal kernel allocation. Because the penalty term adopted here is in proportional to total number of kernels, the constrained optimization is reduced to a ML problem.

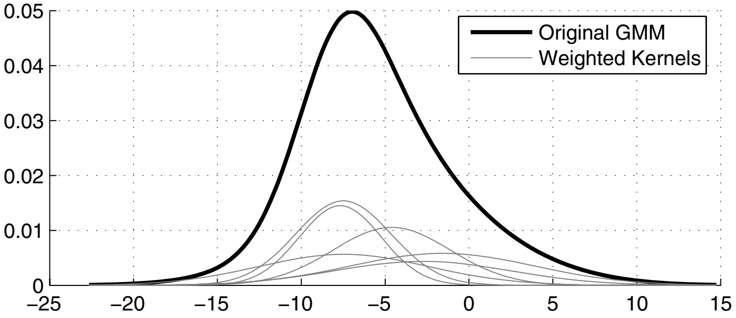


Fig. 1. An example of over-modeled Gaussian mixture

2.1 Likelihood Decomposition

In principle, given an arbitrary m, we need to re-train corresponding HMMs to obtain the likelihoods. However, this may be too expensive to be practical. Therefore, we introduce some reasonable assumptions to simplify the optimization process, and no re-training is needed. A basic assumption adopted is as follows:

Assumption 1: The likelihood of a given state of m kernels is independent of any other states.

Since $\gamma_j(t)$ reflects the state occupancy on the implicit segmentations of the reference transcription, it is reasonable to assume that they only change slightly from one configuration to another. Hence, assumption 1 can also be adopted approximately. Actually, this assumption is based on a intrinsic nature of reference data likelihood: Given an observation, the likelihood of a certain model is independent of other competing models. For other discriminative measures such as posterior probability or mutual information, the property is not valid.

With assumption 1, the total likelihood of the entire database can be decomposed into state likelihoods. Or equivalently, for a given configuration m, we have:

$$L(m) = \sum_{j=1}^{J} L_j(m_j) \tag{3}$$

where $L_j(m_j) = \sum_{t=1}^{T} \gamma_j(t) \log b_j(o_t)$ is the sum of likelihoods of all frames belong to the j^{th} state with m_j Gaussian kernels. In this paper, we refer to it as the *state likelihood function.*

Based on the decomposition, (2) can be simplified to:

$$\hat{m} = \mathrm{argmax}\ _m \Big[\sum_{j=1}^{J} L_j(m_j) - P(\ m) \Big] \quad (\text{s.t.} \sum_{j=1}^{J} m_j = J M_{\mathrm{T}}) \tag{4}$$

Because the panelty terms are pre-defined, we can perform the optimization as a post-process without re-training to obtain the corresponding state likelihoods.

2.2 Likelihood Normalization

Since state occupancies can be affected by Baum-Welch training, we normalize the state likelihoods functions to make the assumption 1 more plausible.

Donote $\mathit{\Gamma}_j(m_j) = \sum_{t=1}^{T} \gamma_j(t) \mid_{m_j}$ the total occupancy of state j with m_j kernels, we obatin average occupancy of first:

$$\bar{\mathit{\Gamma}}_j = \frac{1}{M_{\mathrm{U}}} \sum_{m_j=1}^{M_{\mathrm{U}}} \mathit{\Gamma}_j(m_j) \tag{5}$$

where M_{U} is maximal number of kernel per state used in the training process.

Accordingly, the normalized state likelihood function is then:

$$\bar{L}_j(m_j) = \frac{\bar{\mathit{\Gamma}}_j}{\mathit{\Gamma}_j(m_j)} L_j(m_j) \tag{6}$$

The resultant normalized state likelihood functions are used in the optimization of (4).

3 Step Back Algorithm with Convex Likelihood Functions

Now we are facing the problem of (4), which is a combinatorial optimization. Because the number of possible hypotheses of configuration vector m can be huge, it is very expensive to solve the problem exhaustively. Therefore, we try to simplify the solution by introducing one more assumption:

Assumption 2: Normalized state likelihood of all states are convex, or equivalently:

$$\bar{L}_j(m) - \bar{L}_j(m-1) \geq \bar{L}_j(m+1) - \bar{L}_j(m) \tag{7}$$

The physical meaning is that the marginal contribution of each new Gaussian kernel to the state likelihood does not increase with increasing kernels. We found that the assumption is approximately correct and confirmed experimentally. To illustrate this, we show the normalized state likelihoods as a function of different number of kernels obtained in the training process. The curves are plotted in Figure 2. From the figure we observe that the normalized state likelihood functions are approximatively convex. Note that $P(\ m)$ is proportional to the total number of kernels, the overall criterion is also convex.

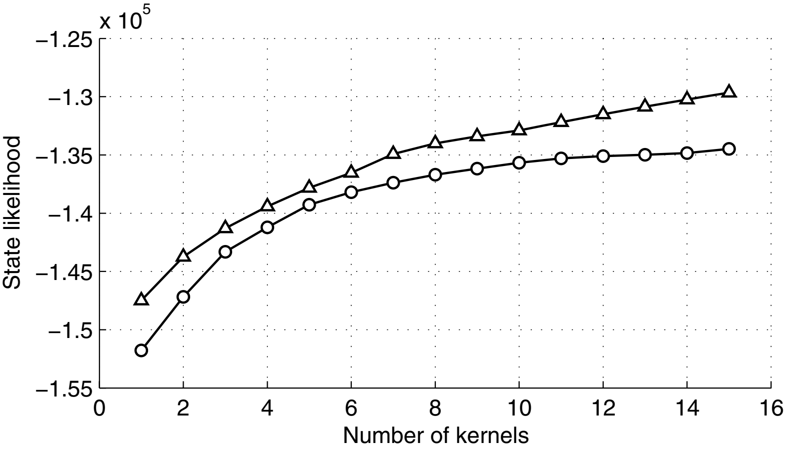


Fig. 2. Examples of convex state likelihood functions

Let M_{U}, M_{L} denote the maximum, minimum number of kernels in each state, respectively, M_{T} denotes the target average number of kernels, where $M_{\text{L}} < M_{\text{T}} < M_{\text{U}}$, we come up with an algorithm to solve (5). Denote $\Delta C_j(m_j) = \bar{L}_j(m_j) - \bar{L}_j(m_j - 1) - (0.5 + d)\log Td$, the psuedo code of the algorithm is shown in Table 1.

In Table 1, the first block is the conventional Baum-Welch training process, where all the state likelihoods are saved. The second block is MDL based kernel allocation. It reduces the model size successively by eliminating one kernel from the state which yields the minimum reduction of normalized likelihood. The algorithm starts with a relatively larger number of kernels for each state and avoids re-training in pruning. Although the algorithm is greedy in nature, it is easy to prove that the optimal configuration is obtained if the two assumptions are valid. If the number of kernels are constrained, the termination condition of the state pruning can be set at: total number of kernels = $J \times M_{\text{T}}$.

Table 1. *Step back algorithm for non-uniform kernel allocation*

```
Initialization:
    Do Baum-Welch training for models with $M_{\mathrm{U}}$ kernels per state and record state likelihood functions $L_j$
    For each state $1 \leq j \leq J$
        Normalize $L_j$ to $L_j$
        Set $m_j$ to $M_{\mathrm{U}}$
Pruning:
    While $\min_{j,m_j>M_{\mathrm{L}}} \Delta C_j(m_j) \leq 0$
        Find the state $i$ with minimal criterion reduction:
            $i = \mathrm{argmin}_{j,m_j>M_{\mathrm{L}}} \Delta C_j(m_j)$
        Decrement $m_i$ and total number of kernels by 1.
Kernel grouping and refinement:
    For each state $1 \leq j \leq J$
        Group the $M_{\mathrm{U}}$ kernels to $m_j$ kernels
    Retrain the models using Baum-Welch algorithm.
```

3.1 Kernel Pruning by Heap Sorting

We apply heap sorting to the kernel pruning process for selecting the state to eliminate a kernel in each step: 1) Heap initialization: the heap is initialized with delta likelihoods $\Delta\tilde{L}_j$ of all the states; 2) State selecting: in each step, the state with minimum delta likelihood is selected from the heap; 3) State refreshing: after pruning, the new delta likelihood of the same state is inserted back into the heap. The complexity of the entire kernel pruning procedure is $O[(M_{\mathrm{U}} - M_{\mathrm{T}})J^2 \log J]$.

3.2 Gaussian Kernel Grouping

The kernel grouping and refinement in the algorithm is performed in two successive steps: First, the M_{U} kernels in each state are clustered into $\hat{m}_j$ groups; second, a new Gaussian model of each class is Baum-Welch re-estimated in the maximaum likelihood sense.

In the clustering stage, Kullback-Leibler divergence [9][10] is used as a measure of kernel similarity. The adopted criterion is to minimize the sum of all the intra-class KLDs weighted by the product of the priors of the two corresponding kernels.

In the merging stage, given the K Gaussian kernels $\mathcal{N}(x; \mu_k, \Sigma_k)$ belongs to the same class and their prior w_k, the parameters of the merged Gaussian model $\mathcal{N}(x; \mu, \Sigma)$ and its prior w can be calculated as:

$$\begin{cases} w = \sum_{k=1}^{K} w_k \\ \mu = \sum_{k=1}^{K} w_k \mu_k \Big/ w \\ \Sigma = \sum_{k=1}^{K} w_k [\Sigma_k + (\mu - \mu_k)(\mu - \mu_k)^\top] \mu_k \Big/ w \end{cases} \tag{8}$$

The re-estimated models are refined by several Baum-Welch iterations in the last step.

4 Experiments

To verify the proposed approaches, we carried out experiments DARPA Naval Resource Management (RM) database and MSRA Mandarin Speech Toolbox (MST) [11] database. The models are trained on the corresponding training sets using 39-dimensional MFCC features.

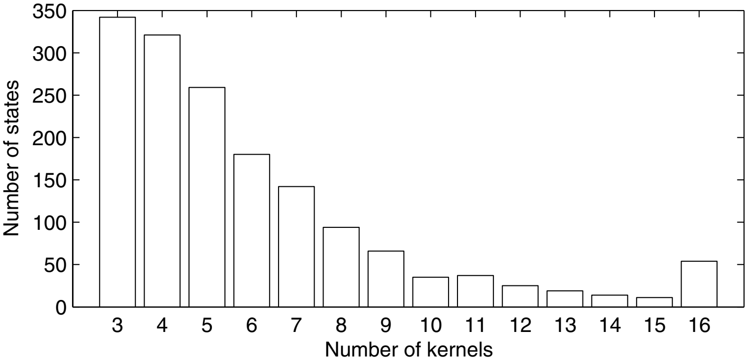


Fig. 3. Histogram of model size (number of kernels) of states

First, we study the model size of states (in terms of number of kernels) after non-uniform kernel allocation. On RM task, with M_{L}=3, M_{U}=16, the statistics of kernel distribution is shown Figure 3. We observe that quite a few models are sufficiently modeled by smaller (than average) number of kernels and extra kernels are allocated to the states which need higher resolutions.

Recognition performance and MDL criterion C in terms of Word Error Rate (WER) is shown in Figure 4, where we adopt $M_{\mathrm{L}} = 1/2M_{\mathrm{T}}, M_{\mathrm{U}} = 1/2M_{\mathrm{T}}$ for parsimonious kernel allocation. It can be observed that parsimonious models outperform baseline models whith a fixed model size per state: For the two databases, when the same total numbers of Gaussian kernels are used, average WER reduction are 11.1% and 5.7%, respectively. The model size can be comprressed by about 35-50% without any loss in accuracy.

We also show the optimal operating points determined by MDL without constraints in number of states. On RM, we can achieve a WER of 3.78% with 5.1 kernels per state, while on Mandarin Speech Toolbox, we can achieve a WER of 20.20% with 14.6 kernels per state. The operating points are significantly better than the baseline, and the model size are compact. But the WERs have not reached optima at these points. This may indicate that for achieving best recognition performance, a more discriminative criterion then ML or MDL is still desirable.

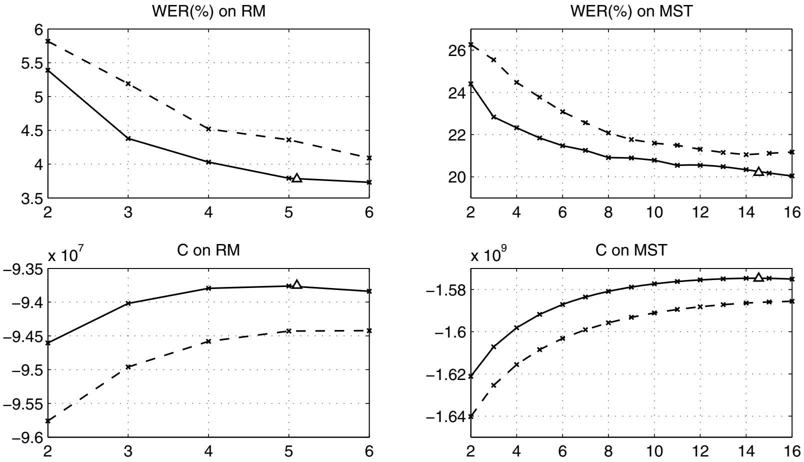


Fig. 4. Recognition performance (WER) and MDL criterion (C) with respect to number of kernels per state. (Dashed lines: uniform kernel allocation; Solid lines: non-uniform kernel allocation; △: optimal model size determined by MDL)

Another observation is when the average number of kernels equals to 2, parsimonious modeling achieves less improvement than other cases. It might be explained by the bimodal distribution of male and female data. Two Gaussian kernels seem to be necessary for characterizing most of the states. Correspondingly, the parsimonious modeling dose not have degrees of freedom to allocate the kernels non-uniformly other than 2 kernels to each state.

5 Conclusions and Discussion

In this paper, we propose a parsimonious modeling algorithm for training HMM by Gaussian kernel allocation. The criteria to be optimized is MDL, and the algorithm is shown to be computational efficient and performance-wise effective. From the experimental results we observe:

1) Parsimonious kernel allocation is promising for model refinement and/or compression.

2) In the ML sense, Gaussian kernel allocation can be solved with a straightforward step-back algorithm, and the resultant models are more parsimonious in describing the overall training data statistics with same number of parameters.

Parsimonious HMMs leads to more compact description of the data. On recognition, the problem can also be investiagted under discriminative criteria, where the non-uniform kernel allocation based upon ML or MDL can be used as a starting point.

References

[1] R. Ariew, "Occam's razor: A historical and philosophical analysis of Ockham's principle of pasimony," *Philosophy*, Champaigh-Urbara, University of Illinois, 1976.

[2] H. Akaike, Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, B.N. Petrov and F. Csake, eds., Budapest: Akademiai Kiado, pp. 267-281, 1973.

[3] G. Schwarz, Estimating the dimension of a model. *Annals of Statistics*, vol. 6(2), pp. 461-464, 1978.

[4] J. Rissanen, *Stochastic complexity in statistical inquiry.* Word Scientific Publishing Company, 1989.

[5] X. B. Li, F. K. Soong, T. A. Myroll, R. H. Wang, "Optimal Clustering and Non-uniform Allocation of Gaussian Kernels in Scalar Dimension for HMM Compression," in *Proc. ICASSP05*, vol. 1, pp. 669-672, 2005.

[6] J. Takami, S. Sagayama, "A Successive State Splitting Algorithm for Efficient Allophone Modeling," in *Proc. ICASSP-92*, volume I, pp. 573-576, 1992

[7] X. D. Huang et al., "The SPHINX-II speech recognition system: An overview," *Comput. Speech Language*, vol. 2, pp. 137-148, 1993.

[8] L. E. Baum and J. A. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bulletin of American Mathematical Society*, vol. 41, pp. 360-363, 1970.

[9] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *Ann. Math. Stat.*, 22: 79-86, 1951.

[10] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley Interscience, NewYork, NY, 1991.

[11] E. Chang, Y. Shi, J.-L. Zhou and C. Huang, "Speech lab in a box: A Mandarin speech toolbox to jumpstart speech related research toolbox," *Eurospeech2001*, pp. 2799-2782.

Consistent Modeling of the Static and Time-Derivative Cepstrums for Speech Recognition Using HSPTM

Yiu-Pong Lai and Man-Hung Siu

Department of Electronic and Computer Engineering,
The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
harry@ust.hk, eemsiu@ust.hk

Abstract. Most speech models represent the static and derivative cepstral features with separate models that can be inconsistent with each other. In our previous work, we proposed the hidden spectral peak trajectory model (HSPTM) in which the static cepstral trajectories are derived from a set of hidden trajectories of the spectral peaks (captured as spectral poles) in the time-frequency domain. In this work, the HSPTM is generalized such that both the static and derivative features are derived from a single set of hidden pole trajectories using the well-known relationship between the spectral poles and cepstral coefficients. As the pole trajectories represent the resonance frequencies across time, they can be interpreted as formant tracks in voiced speech which have been shown to contain important cues for phonemic identification. To preserve the common recognition framework, the likelihood functions are still defined in the cepstral domain with the acoustic models defined by the static and derivative cepstral trajectories. However, these trajectories are no longer separately estimated but jointly derived, and thus are ensured to be consistent with each other. Vowel classification experiments were performed on the TIMIT corpus, using low complexity models (2-mixture). They showed 3% (absolute) classification error reduction compared to the standard HMM of the same complexity.

1 Introduction

Cepstral-based Hidden Markov models (HMMs) are widely used in speech recognition. Most HMMs are frame-based that assume conditional independence between frames. To capture the temporal dependence of speech, cepstral time-derivative features, often also called the dynamic features, are added to augment the "static" cepstral coefficients. These static and dynamic features are typically considered as separate features and no constraint is imposed to ensure the consistence between the trajectories inferred by them. Let us consider the case of a simple multi-state HMM with single diagonal Gaussians as the observation distributions. Suppose the observation space includes 10 static cepstral coefficients and 10 first order derivatives and the Gaussian means are non-zero for all features. Then, the static portion of the model infers a piece-wise constant cepstral trajectory. On the other

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 303–314, 2006.

hand, the non-zero derivatives infer a linearly changing cepstral trajectory which is inconsistent with the trajectory inferred by the static coefficients. If we consider the HMMs as a generative model, many of the generated features (including both static and derivatives) do not correspond to any physical cepstral sequence.

Alternatives were proposed to capture the temporal dependence such as the segmental models [1,2,3,4,5,6]. Although these models make fewer assumptions about the correlation between adjacent frames and can improve the recognition performance, dynamic features are again used to augment the static features to obtain better performance. Thus, the model inconsistency issue remains.

Because the dynamic features are functions of the static features, they are really dependent variables instead of independent variables. The fact that many of the generated feature sequences do not match with any physical cepstral sequence implies that the models are assigning probability to a portion of the feature space that in fact should have zero probability. In [7], the trajectory HMM was proposed that treats the dynamic features as a regression function of the static features with a “corrected” likelihood function. Because the dynamic features depend on the static coefficients across a fixed time span, this likelihood function can be evaluated when the state sequence for the whole sentence is known. This has been shown to work well in both recognition and synthesis but the parameters in an individual modeling unit depend on others so that a large set of linear equations have to be solved in model training.

Instead of correcting the likelihood, another approach to handling model consistency is by generating both the static and time-derivative model from a single set of parameters. The spectral domain is a natural choice because the cepstra is in fact a function of spectral parameters. In [8], vocal tract resonance (VTR) tracks are extracted from the speech acoustics using a stochastic process with continuity constraints in the spectral domain and transformed into the cepstral domain for estimating phonemic models. The advantage is that these tracks are constrained not only within a phonetic unit but also continuous across different units. This is particularly useful for capturing co-articulation effects. Biases, separately estimated for both the static and dynamic features, are added to improve the modeling accuracy. While the cepstral models derived from the VTR are consistent, the addition of the bias again creates an inconsistent model.

Based on the fact that the cepstral trajectories are driven by the spectral parameters, the hidden spectral peak trajectory model (HSPTM) is proposed in our previous work to represent the time-frequency characteristic of speech using the trajectory of all-pole representation of spectrum [9]. These trajectories are modeled as polynomial functions across time similar to the polynomial segment models [10]. Different from [8], the spectral peak trajectories are estimated implicitly using the static cepstral coefficients on a phoneme-by-phoneme basis. The original motivation is to have a more parsimonious representation of the acoustic model. The HSPTM, however, can be extended to derive the time-derivative features. As the polynomials are used to represent the spectral peak trajectories in HSPTM, the time-derivative models in cepstral domain are easily generated from the spectral peaks trajectory parameters. More importantly, within each acoustic unit, these time-derivative models are consistent with the static model in cepstral domain because they are generated from the same set of underlying parameters. Contrast with modeling the VTR, the spectral peaks only capture the spectral-time characteristic and have no prior physical meaning in the acoustic although they can capture the formant in vowels as the formants carry most energy in voiced speech. But in other phonetic units, the spectral peaks serve as band-pass filters that capture the resonances of sound production articulators. Therefore, it may not be necessary to use the bias in the cepstral model and consistence of static and time-derivative trajectories can be maintained.

Besides modeling the temporal information of speech, two modifications of HSPTM will also be presented. First, it is well known that single Gaussian is not enough to capture the statically variation of speech. Therefore, Gaussian mixtures are usually used in recognition system and this can also be applied in HSPTM to improve the recognition accuracy. Second, the phonetic units can be represented using multiple trajectories. According to the results reported in [10], using multiple segments is preferable in speech recognition. Although this would reduce the modeling power of temporal information, the sub-phonetic boundary can be varied so that it is more flexible to model the variation in the phonetic units when linear normalization is used in the trajectories model.

In this paper, we will formulate the time-derivative cepstral model using spectral peak trajectories and discuss the use of the sub-phonetic HSPTM in Section 2. In Section 3, the HSPTM parameter estimation will be presented together with the extension to mixture model. Since our purpose is to illustrate the feasibility of a consistent HSPTM and to simplify the experimental complexity, only low complexity models with two mixture components were used in our experiments. The experimental results on vowel classification will be reported in section 4. The paper will be summarized in section 5.

2 The Hidden Spectral Peak Trajectory Model

Using the relationship between the cepstral coefficients and the spectral peaks, the HSPTM can capture the coarse shapes of frequency-time characteristic [9]. Since the temporal information of speech is also important for recognition, especially for phonetic units with short duration, the time-derivative features can be used as additional information. Under HSPTM, the spectral peaks for each acoustic unit are represented using polynomial functions and can easily be extended to generate both static and the time-derivative models in the cepstral domain. The time-derivative model generated is guaranteed to be consistent with the static model. As the dynamic cepstral features are now part of the observations in deriving the spectral peak parameters, they provided information on a wider time span such that higher order polynomial coefficients can be estimated even on segments with short durations, such as in the case of only a few frames in the sub-phonetic units. More detail will be discussed below.

Let us consider an all-pole model with $2p$ poles. The spectral transfer function $H(z)$ can be written as

$$H(z) = \frac{G}{\prod_{i=1}^{p}(1 - z_i z^{-1})(1 - z_i^* z^{-1})}, \tag{1}$$

where G is the gain and, z_i and z_i^* denote a root and its complex conjugate. The i_{th} root can be expressed as

$$z_i = e^{-\pi B_i + j2\pi F_i}, \quad 1 \leq i \leq p, \tag{2}$$

where B_i, F_i $(0 < F_i < \frac{1}{2})$ are its bandwidth and center frequency .

To capture the time-dependent characteristics, the spectral peaks trajectories are represented by polynomial time functions. Denote the bandwidth and center frequency of the i_{th} spectral peak trajectory at time frame n as $B_i(n)$ and $F_i(n)$, which are polynomial functions with a normalized time scale as described in [1]. For a length-N linear segment, $F_i(n)$ and $B_i(n)$ are then defined as

$$F_i(n) = \sum_{j=0}^{J} \omega(i,j) \left(\frac{n-1}{N-1} \right)^j, \tag{3}$$

$$B_i(n) = \sum_{j=0}^{J} \beta(i,j) \left(\frac{n-1}{N-1} \right)^j, \tag{4}$$

where $\omega(i,j)$ and $\beta(i,j)$ are the j^{th} coefficients of a J-th order polynomial for the i^{th} trajectories. To be consistent with formant in the voiced sounds, the spectral peak trajectories are assumed to be in order and cannot cross each other.

The cepstral coefficients can be computed by taking the inverse z-transforms on the logarithm of the transfer function [11]. The k^{th} cepstral coefficient is given as

$$c_k = \frac{2}{k} \sum_{i=1}^{p} e^{-\pi k B_i} \cos(2\pi k F_i), \quad k > 0. \tag{5}$$

The cepstral trajectories $\mu_\phi(n,k)$ at time index n, and their first and second order time derivatives, $\mu'_\phi(n,k)$ and $\mu''_\phi(n,k)$, can be written as functions of the spectral peak trajectories given by,

$$\mu_\phi(n,k) = \frac{2}{k} \sum_{i=1}^{p} \psi_{\phi,i}(n,k) \tag{6}$$

$$\mu'_\phi(n,k) = -2\pi \sum_{i=1}^{p} \left[\psi_{\phi,i}(n,k) B'_i(n) + 2\zeta_{\phi,i}(n,k) F'_i(n) \right] \tag{7}$$

$$\begin{aligned}\mu''_\phi(n,k) = 2\pi \sum_{i=1}^{p} \{ &\psi_{\phi,i}(n,k) \left[\pi k (B'_i(n))^2 - 4\pi k (F'_i(n))^2 - B''_i(n) \right] \\ &+ \zeta_{\phi,i}(n,k) \left[4\pi k F'_i(n) B'_i(n) - 2F''_i(n) \right] \}\end{aligned} \tag{8}$$

where

$$\psi_{\phi,i}(n,k) = e^{-\pi k B_i(n)}\cos(2\pi k F_i(n)), \tag{9}$$
$$\zeta_{\phi,i}(n,k) = e^{-\pi k B_i(n)}\sin(2\pi k F_i(n)), \tag{10}$$

and $X_i'(n)$ and $X_i''(n)$ are the first and second derivatives of function X against n. In equations 7 and 8, the mean trajectories of the dynamic features are generated from the functions of the spectral peak trajectories. This guarantees its consistency with the static trajectory.

The cepstral trajectories depend on the segment length. Denote $\boldsymbol{O} = [\boldsymbol{o}(1), \ldots, \boldsymbol{o}(N)]$ to be a length N sequence of $3D$-dimensional feature vector $\boldsymbol{o}(n) = [o(n)^T, \Delta o(n)^T, \Delta^2 o(n)^T]^T$, that includes the static D-dimensional cepstral vector and their first and second order derivatives. These features are assumed to be generated by a set of time-varying mean trajectories $\boldsymbol{\mu}_\phi(n)$ of model ϕ with a zero-mean residue $\mathbf{e}_\phi(n)$ [1], with variance $\boldsymbol{\Sigma}_\phi$ such that

$$\boldsymbol{\mu}_\phi(n) = [\mu_\phi(n,1)\ldots\mu_\phi(n,D), \mu_\phi'(n,1)\ldots\mu_\phi'(n,D), \mu_\phi''(n,1)\ldots\mu_\phi''(n,D)]^T \tag{11}$$

$$\boldsymbol{o}(n) = \boldsymbol{\mu}_\phi(n) + \mathbf{e}_\phi(n) \;\text{ for } 1{\le}n{\le}N. \tag{12}$$

Using Equation 12, the log likelihood can be written as,

$$\begin{aligned}\mathcal{L}(\boldsymbol{O}|\lambda_\phi) = \log P(\boldsymbol{O}; \omega_\phi, \beta_\phi, \boldsymbol{\Sigma}_\phi) &= -\frac{1}{2}\sum_{n=1}^{N}[-3D\log(2\pi) + \log(|\boldsymbol{\Sigma}_\phi|) \\ &+ (\boldsymbol{o}(n) - \boldsymbol{\mu}_\phi(n))^T \boldsymbol{\Sigma}_\phi^{-1}(\boldsymbol{o}(n) - \boldsymbol{\mu}_\phi(n))]\end{aligned} \tag{13}$$

Noted that, once the parameters of the spectral peak trajectories are trained, the mean trajectory $\boldsymbol{\mu}_\phi$ only depends on the segment length so that all likelihood computation can be carried out in cepstral domain. In this sense, the spectral peak trajectories are "hidden" and only the cepstral representation are used.

2.1 Sub-phonetic Modeling Units

In traditional segment models, a single segment is usually used to model each phonetic unit [12]. This has several advantages. It reduces the number of segment boundaries needed to be searched during recognition and ensures the continuity and smoothness of the trajectory within each phonetic unit. In addition, it allows joint modeling of all the observations within each phonetic unit.

However, this may not be flexible enough to capture temporal or acoustic variations. For example, co-articulatory effect may affect the spectral characteristic in the boundary of phonetic units. Compared to the HMM, each phonetic unit is usually modeled by at least three separate states which can be considered as the left boundary region affected by left context, the stationary region and right boundary region affected by the right context. With single trajectory per segment, variations caused by different left contexts can only be represented by adding new trajectories for the whole phonetic unit even though the stationary and right boundary regions are more or less the same. This also can be problematic for parameter sharing, such as in context-dependent modeling. Furthermore, the time-normalization used in these segments in effect applies a uniform sampling of the trajectory across each unit. With phonetic units, it does not allow duration variations of the different boundary regions relative to each other. That is, an instance of a longer phonetic unit would extend both the boundary regions and the stationary regions to the same extent, even though it may be more likely that the speaker has elongated the stationary region.

To address these issues, sub-phonetic units can be used. By dividing each phoneme into different independent sub-phonetic units represented by different trajectories, more flexibility in capturing the co-articulatory effect and for parameter sharing is added while some within segment correlation are still captured. In this paper, three sub-phonetic segments are used to represent each phoneme.

2.2 Example of Data Fitting

How well does HSPTM fit the data in cepstral domain? To illustrate the differences in speech trajectory fitting in the cepstral domain, HMMs and HSPTMs were trained with a single instance of the vowel “ae”. For HMM, a 3-state, left-to-right topology was used with single Gaussian per state. Similarly, HSPTM used three sub-phonetic units with six spectral peak trajectories represented by

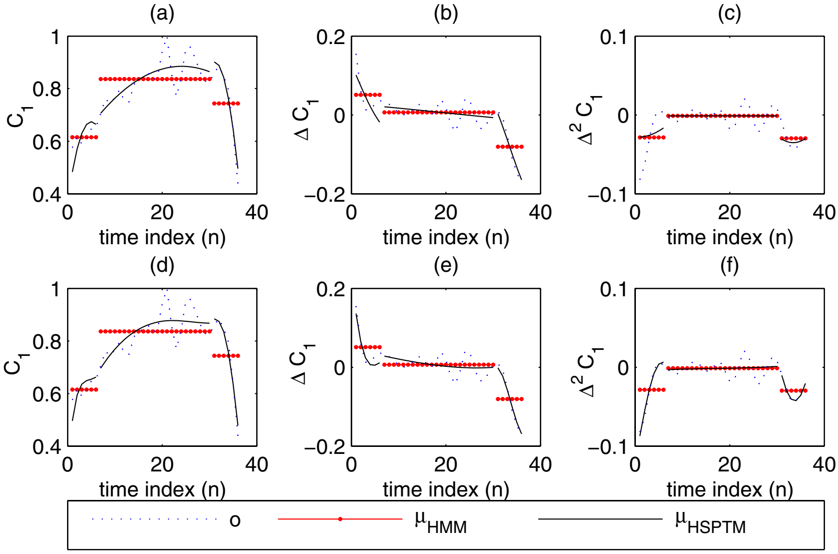


Fig. 1. Mean trajectories of the first cepstral coefficient of HMM and HSPTM of the vowel 'ae'. (a)-(c)second-order polynomial function, (d)-(f)forth-order polynomial function.

quadratic polynomials for both the center frequency and bandwidth. To simplify the comparison, the HSPTM used the HMM state alignment. The mean trajectories of the first cepstral coefficient (c_1) and its time-derivatives are shown separately in Figure 1. The dotted lines represent the coefficients actual cepstral observations, the solid lines with dots show the HMM mean trajectories and the solid lines represent the HSPTM fitted mean trajectory.

It can be seen from Figure 1 that the HMM trajectories are piece-wise stationary such that they are unable to accurately fit both static and time-derivative observations. For the HSPTM, while the cepstral trajectories were derived from the spectral peak trajectories in the time-frequency domain, the cepstral trajectories can represent more accurately the temporal relationship of both the static and dynamic coefficients simultaneously with smaller fitting error compared with HMMs. This is particularly prominent in the third state.

One may notice that the HSPTM is also not representing the time-derivative coefficients in first state as shown in Figure 1(b) and 1(c). Given that this is a consistent model in that the static and derivative cepstral trajectories were jointly derived from a single set of spectral peak trajectories, an accurate representation of the static trajectory may limit the accuracy of the derivative trajectories. This also suggests that the spectral peak trajectories are not sufficiently accurate with quadratic polynomials. To verify this conjecture, HSPTMs with forth order polynomials were trained and the resulting cepstral trajectories are shown in Figure 1(d)-(f). It can be seen that the fit in the first sub-phonetic units is significantly improved and the trajectories in the third state are nearly perfectly on top of observed data.

Similar phenomena are also observed for other cepstral coefficients. It is also interesting to notice that the trajectories for the second and the fourth order polynomial are similar in the center region which suggests that higher order polynomials should be used only when the changes of the coefficients are large, or during more transient phonetic units. The modeling flexibility depends on the number of sub-phonetic units and the polynomial order. Since both approaches would increase the model complexity, and a balance between them would be important.

2.3 Number of Parameters

How many parameters are in one HSPTM sub-phonetic segment as compared to an HMM state? Since both are assumed to be Gaussian distributed in the cepstral domain, they use the same number of parameters for the convariance matrix. For the mean trajectory, this depends on the number of hidden spectral peak tracks and the polynomial order. For K spectral-peak tracks with R-th order polynomial, the number of parameter is simply $2K \times (R + 1)$ because two polynomials are needed with one for the center frequency and one for the bandwidth and they have the same polynomial order. Note that the number of HSPTM parameters for the mean trajectory is independent of the number of time derivatives, nor the number of cepstral coefficients used. However, we can see in Figure 1 as well as in the next section that the accuracy of the spectral peaks are measured by the fit in the cepstral domain. Thus, the number of cepstral coefficient used would indirectly affects the accuracy of the hidden spectral peaks.

Consider the widely used 39 dimensional feature vectors, with the 12 cepstral coefficients plus energy, and their first and second derivatives. Suppose six hidden spectral peaks, represented by quadratic polynomial for both center frequency and bandwidth while the frame energy is represented as a flat trajectory. The total number of HSPTM parameters would be $2 \times 6 \times 3 = 36$ for the hidden spectral peaks plus 3 for the frame energy resulting in 39 parameters. This is exactly the same as the HMM.

3 Model Parameters Estimation

The HSPTM parameters include the variances $\boldsymbol{\Sigma}$'s, and the polynomial coefficients $\omega(i,j)$'s and $\beta(i,j)$'s. It can be seen that in Equations 6-8, every point in the both static and time-derivative cepstral trajectories depends on all the polynomial coefficients through the terms in Equations 9 and 10. However, the non-linearity in these function creates a difficulty in the model estimation and no closed form solution of the maximum-likelihood parameter estimation problem can be found. Instead, an iterative gradient descent method has to be used in the estimation of the polynomial coefficients of frequency and bandwidth trajectories. The variance are also estimated iteratively using the updated parameters of the polynomial coefficients.

Denote the features R training tokens of model ϕ as $\{\boldsymbol{O}_1, \boldsymbol{O}_2, \ldots, \boldsymbol{O}_R\}$ and the length of $\boldsymbol{O}_R$ as N_r. Because the mean trajectories depend on the segment length, the training token are grouped according to the length. The update equations for $\omega(i,j)$ and $\beta(i,j)$ are:

$$\begin{aligned}\hat{\omega}(i,j) &= \omega(i,j) - \epsilon \sum_{r=1}^{R} \frac{\partial \mathcal{L}(\boldsymbol{O}_r|\lambda_\phi)}{\partial \omega(i,j)} = \omega(i,j) - \epsilon \sum_{N=1}^{\max\{N_r\}} \sum_{\forall r: N_r = N} \frac{\partial \mathcal{L}(\boldsymbol{O}_r|\lambda_\phi)}{\partial \omega(i,j)} \\ &= \omega(i,j) - \epsilon \sum_{N=1}^{\max\{N_r\}} \sum_{\forall r: N_r = N} \sum_{n=1}^{N_r} \left[\frac{\partial \boldsymbol{\mu}_\phi(n)^T}{\partial \omega(i,j)} \boldsymbol{\Sigma}_\phi^{-1} (\boldsymbol{o}_r(n) - \boldsymbol{\mu}_\phi(n)) \right] \end{aligned} \quad (14)$$

$$\begin{aligned}\hat{\beta}(i,j) &= \beta(i,j) - \epsilon \sum_{r=1}^{R} \frac{\partial \mathcal{L}(\boldsymbol{O}_r|\lambda_\phi)}{\partial \beta(i,j)} = \beta(i,j) - \epsilon \sum_{N=1}^{\max\{N_r\}} \sum_{\forall r: N_r = N} \frac{\partial \mathcal{L}(\boldsymbol{O}_r|\lambda_\phi)}{\partial \beta(i,j)} \\ &= \beta(i,j) - \epsilon \sum_{N=1}^{\max\{N_r\}} \sum_{\forall r: N_r = N} \sum_{n=1}^{N_r} \left[\frac{\partial \boldsymbol{\mu}_\phi(n)^T}{\partial \beta(i,j)} \boldsymbol{\Sigma}_\phi^{-1} (\boldsymbol{o}_r(n) - \boldsymbol{\mu}_\phi(n)) \right] \end{aligned} \quad (15)$$

where ϵ is the learning rate.

To find the derivatives of the mean trajectories against the parameters, $\frac{\partial \boldsymbol{\mu}_\phi(n)^T}{\partial \omega(i,j)}$ and $\frac{\partial \boldsymbol{\mu}_\phi(n)^T}{\partial \beta(i,j)}$, each dimension can be separately computed and expressed as functions defined in Equations 9 and 10 and derivatives of Equations 3 and 4.

$$\frac{\partial \mu_\phi(n,k)}{\partial \omega(i,j)} = -4\pi \zeta_{\phi,i}(n,k) \left(\frac{\partial F_i(n)}{\partial \omega(i,j)} \right), \quad (16)$$

$$
\begin{aligned}
\frac{\partial \mu_\phi(n,k)}{\partial \beta(i,j)} &= -2\pi \psi_{\phi,i}(n,k)(\frac{\partial B_i(n)}{\partial \beta(i,j)}), && (17)\\
\frac{\partial \mu'_\phi(n,k)}{\partial \omega(i,j)} &= 2\pi \Big\{ \zeta_{\phi,i}(n,k) \Big[2\pi k B'_i(n) \tfrac{\partial F_i(n)}{\partial \omega(i,j)} - 2 \tfrac{\partial F'_i(n)}{\partial \omega(i,j)} \Big] \\
&\quad + \psi_{\phi,i}(n,k) \Big[-4\pi k F'_i(n) \tfrac{\partial F_i(n)}{\partial \omega(i,j)} \Big] \Big\}, && (18)\\
\frac{\partial \mu'_\phi(n,k)}{\partial \beta(i,j)} &= 2\pi \Big\{ \psi_{\phi,i}(n,k) \Big[\pi k B'_i(n) \tfrac{\partial B_i(n)}{\partial \beta(i,j)} - \tfrac{\partial B'_i(n)}{\partial \beta(i,j)} \Big] \\
&\quad + \zeta_{\phi,i}(n,k) \Big[2\pi k F'_i(n) \tfrac{\partial B_i(n)}{\partial \beta(i,j)} \Big] \Big\}, && (19)\\
\frac{\partial \mu''_\phi(n,k)}{\partial \omega(i,j)} &= 2\pi \Big\{ \psi_{\phi,i}(n,k) \Big[\pi k (8\pi k F'_i(n) B'_i(n) - 4F''_i(n)) \tfrac{\partial F_i(n)}{\partial \omega(i,j)} \\
&\quad - 8\pi k (F'_i(n)) \tfrac{\partial F'_i(n)}{\partial \omega(i,j)} \Big] + \zeta_{\phi,i}(n,k) \big[\pi k (-2\pi k (B'_i(n))^2 + 8\pi k (F'_i(n))^2 \\
&\quad +2B''_i(n)) \tfrac{\partial F_i(n)}{\partial \omega(i,j)} + \ 4\pi k B'_i(n) \tfrac{\partial F'_i(n)}{\partial \omega(i,j)} - 2 \tfrac{\partial F''_i(n)}{\partial \omega(i,j)} \Big] \Big\}, && (20)\\
\frac{\partial \mu''_\phi(n,k)}{\partial \beta(i,j)} &= 2\pi \Big\{ \zeta_{\phi,i}(n,k) \Big[\pi k (-4\pi k F'_i(n) B'_i(n) + 2F''_i(n)) \tfrac{\partial B_i(n)}{\partial \beta(i,j)} \\
&\quad + \ 4\pi k F'_i(n) \tfrac{\partial B'_i(n)}{\partial \beta(i,j)} \Big] + \psi_{\phi,i}(n,k) \big[\pi k (-\pi k (B'_i(n))^2 + 4\pi k (F'_i(n))^2 \\
&\quad +B''_i(n)) \tfrac{\partial B_i(n)}{\partial \beta(i,j)} + \ 2\pi k (B'_i(n)) \tfrac{\partial B_i(n)}{\partial \beta(i,j)} - \tfrac{\partial B''_i(n)}{\partial \beta(i,j)} \Big] \Big\}. && (21)
\end{aligned}
$$

Although the expressions are quite complicated, they are general to any order of polynomial. Moreover, they are in terms of $\psi_{\phi,i}(n,k)$, $\zeta_{\phi,i}(n,k)$ and the derivatives of $F_i(n)$ and $B_i(n)$ and some of these terms can be cached to reduce computation.

The re-estimation for the covariance using the updated polynomial coefficients and is given by

$$
\hat{\boldsymbol{\Sigma}}_\phi = \frac{\sum\limits_{N=1}^{\max\{N_r\}} \sum\limits_{\forall r : N_r = N} \sum\limits_{n=1}^{N} \left(\boldsymbol{o}(n) - \boldsymbol{\mu}_\phi(n)\right) \left(\boldsymbol{o}(n) - \boldsymbol{\mu}_\phi(n)\right)^T}{\sum\limits_{N=1}^{\max\{N_r\}} \sum\limits_{\forall r : N_r = N} N} \qquad (22)
$$

One advantage of using the derivative coefficients is that they provide additional independent information for the estimation, so that the minimum number of frames required to estimate the polynomial coefficient can be reduced. This could be important when sub-phonetic units are used and the segment may consist of only one or two frames.

It is well known that a single Gaussian may not be sufficient to capture the statistical variations of the data in HMM and in segmental-based model. Thus, a mixture model was widely used in HMM as well as polynomial trajectory model [10]. Similar technique can also be applied to HSPTM to capture the variation of speakers or speaking styles. Similar to other segmental models, mixture hopping is not allowed in HSPTM. Thus, discontinuity of spectral peak trajectories can only occur at the boundary of modeling units.

From Equation 13, the likelihood of a segment over a K-mixture model is given as

$$P(\boldsymbol{O}|\lambda_\phi) = \prod_{n=1}^{N} \sum_{k=1}^{K} w_{\phi,k} N\left(\boldsymbol{o}(n); \boldsymbol{\mu}_{\phi,k}(n), \boldsymbol{\Sigma}_{\phi,k}\right)$$

where $w_{\phi,k}$ is the mixture weight, $\boldsymbol{\mu}_{\phi,k}$ is the trajectory mean and $\boldsymbol{\Sigma}_{\phi,k}$ is the covariance for the k^{th} mixture component of model ϕ. The update equation of the mixture weight $\hat{w}_{\phi,k}$ is calculated by the model parameters of the previous iteration as follows:

$$\hat{w}_{\phi,k} = \frac{\sum\limits_{N=1}^{\max\{N_r\}} N \sum\limits_{\forall r:N_r=N} \gamma_{\phi,k}(\boldsymbol{O}_r)}{\sum\limits_{m=1}^{K} \sum\limits_{N=1}^{\max\{N_r\}} N \sum\limits_{\forall r:N_r=N} \gamma_{\phi,m}(\boldsymbol{O}_r)} \tag{23}$$

where $\gamma_{\phi,k}$ is the mixture posterior given by

$$\gamma_{\phi,k}(\boldsymbol{O}_r) = \frac{w_{\phi,k} P(\boldsymbol{O}_r|\lambda_{\phi,k})}{\sum_{m=1}^{K} w_{\phi,m} P(\boldsymbol{O}_r|\lambda_{\phi,m})} \tag{24}$$

Because the posterior probabilities γ's are estimated on a per segment basis, they are weighted by the segment duration N.

4 Experiments

Vowel classification experiments were conducted in TIMIT database to evaluate the effectiveness of HSPTM. The experimental setup was similar to those reported by Gish and Fukada [13,12] in which classification of 16 vowels sounds, including 13 monothongs and 3 diphthongs was considered. The only difference is that only the 'sx' and 'si' utterances were used for training and evaluation in our experiments with the 'sa' sentences excluded. This is motivated by the fact that the 'sa' sentences appears in both training and testing such that their phonetic contexts are always observed. Thus, they may be slightly easier to classify. Some of our preliminary results have also shown that including the 'sa' sentences can increase classification accuracy slightly. Without the 'sa' sentences, our training and testing sets consisted of 31864 and 11606 tokens respectively.

The speech was recorded at a sampling rate of 16kHz in the TIMIT corpus. The frontend features were generated using the HTK [14] and included 12 PLP cepstral features (c_1 through c_{12}), the frame energy and their first and second order time derivatives. As is typically with TIMIT experiments, an analysis windows of 25ms with 5ms shifts were used. Cepstral mean normalization was also applied although not straightly necessary because of the consistent recording in the TIMIT corpus.

3-state, left-to-right topology was used to represent each vowels under HMM. Similarly, 3 sub-phonetic units were used in HSPTM. The frame energy was assumed to be conditionally independent and stationary within a modeling unit.

That is, it is represented by a constant within each HSPTM segment. However, because mixture hopping is not allowed in HSPTM, this is still different from HMM. For both HMM and HSPTM, residue covariance is assumed to be diagonal.

Two different HSPTM experiments were performed. One used HMM state alignment while the other optimizes the sub-phonetic segment boundaries. This can be achieved either by searching around HMM state alignment for larger tasks or by exhaustive search for smaller tasks.

Table 1. Vowel Classification Rate

Model	Number of mixtures	
	1	2
HMM	55.02%	56.32%
HSPTM (without alignment training)	55.41%	56.39%
HSPTM (with alignment training)	56.33%	59.11%

Table 1 shows the vowel classification performance of HSPTM and HMM. The second and third rows tabulate the performance of HSPTM using the HMM state alignment and optimized alignment respectively. Comparing the first and second rows shows that HSPTM is only slightly better than HMMs with the gain decreases in two-mixture experiment.

Comparing the second and third rows, optimizing the segment alignment actually significantly improved performance for both one- and two-mixture experiments. Because HSPTM is co-optimized for both static and time-derivative coefficients, the sub-phonetic HSPTM may define a different sub-phonetic unit from an HMM state. The results in the third row show the improved performance of HSPTM for generating the cepstral trajectories over using HMMs. The gain can come from the combination of using a consistent model as well as the use of segments which has been shown to outperform HMMs in vowel classification tasks.

5 Conclusion

In this paper, we explored the consistence between the static and derivative features. We proposed a consistent model by modifying the hidden spectral peak trajectory model (HSPTM) such that both the static and time-derivative cepstral trajectories can be derived from a single set of hidden spectral peak tracks. To improve the model's flexibility in capturing acoustic variations, we also introduced the sub-phonetic HSPTM with mixtures in which multiple segments are used to represent each phonetic unit and each segment can be a mixture of trajectories. In vowel classification experiments on the TIMIT corpus, we showed that the consistent HSPTM out-performed the traditional HMM.

From the experiments, we observed that optimal "state" alignments are important and one of the our future work is to develop efficient HSPTM alignment algorithms.

References

1. Gish, H., Ng, K.: A segmental speech model with applications to word spotting. Proc. of the IEEE Inter. Conf. on Acoust., Speech and Signal Proc. (1993) 447–450
2. Homes W., Russell M.: Probabilistic-trajectory segmental HMMs. Computer Speech and Language. **13** (1999) 3–37
3. Siu M., Iyer R., Gish H., Quillen C.: Parametric trajectory mixtures for lvcsr. Proc. of the Inter. Conf. on Spoken Language Processing. (1998)
4. Goldberger J., Burshtein D., Franco H.: Segmental modeling using a continuous mixture of nonparametric models. IEEE Trans. on Speech and Audio Processing. **7** (1999) 262–271
5. Deng L., Aksmanovic M., Sun X., Wu C.: Speech recognition using hidden markov models with polynomial regression functions as nonstationary states. IEEE Trans. on Speech and Audio Processing. (1994) 507–520
6. Li C., Siu M.: An efficient incremental likelihood evaluation for polynomial trajectory model using with application to model training and recognition. Proc. of the IEEE Inter. Conf. on Acoust., Speech and Signal Proc. (2003) 756–759.
7. Zen H., Tokuda K., Kitamura T.: Trajectory modeling based on HMMs with the explicit relationship between static and dynamic features. Proc. of the IEEE Inter. Conf. on Acoust., Speech and Signal Proc. (2004) 837–840.
8. Deng L., Yu D., Acero A.: A bidirectional target-filtering model of speech coarticulation and reduction: Two-stage implementation for phonetic recognition.IEEE Trans. on Speech and Audio Processing. **14** (2006) 256–265
9. Lai Y., Siu M.: Hidden spectral peak trajectory model for phone classification. Proc. of the IEEE Inter. Conf. on Acoust., Speech and Signal Proc. (2004) 17–21
10. Au Yeung S., Li C., Siu M.: Sub-phonetic polynomial segment model for large vocabulary continuous speech recognition Proc. of the IEEE Inter. Conf. on Acoust., Speech and Signal Proc. (2005) 193–196
11. Huang X. D., Acero A., Hon H. W.: Spoken language processing: A guide to theory, algorithm and system development. Upper Saddle River, New Jersey: Prentice Hall Inc. (2000)
12. Fukada T., Sagisaka Y., Paliwal K. K. Model parameter estimation for mixture density polynomial segment models Proc. of the IEEE Inter. Conf. on Acoust., Speech and Signal Proc. (1997) 1403–1406
13. Gish H., Ng K. Parametric trajectory models for speech recognition: Proc. of the Inter. Conf. on Spoken Language Processing. (1996) 466–469
14. Young S., Evermann G., Kershaw D., Moore G., Odell J., Ollason D., Valtchev V., Woodland P.: The HTK book. (2001)

Vector Autoregressive Model for Missing Feature Reconstruction

Xiong Xiao[1,2], Haizhou Li[1,2], and Eng Siong Chng[1]

[1] School of Computer Engineering, Nanyang Technological University, Singapore
[2] Institute for Infocomm Research, Singapore
xiao0007@ntu.edu.sg, hli@i2r.a-star.edu.sg, aseschng@ntu.edu.sg

Abstract. This paper proposes a Vector Autoregressive (VAR) model as a new technique for missing feature reconstruction in ASR. We model the spectral features using multiple VAR models. A VAR model predicts missing features as a linear function of a block of feature frames. We also propose two schemes for VAR training and testing. The experiments on AURORA-2 database have validated the modeling methodology and shown that the proposed schemes are especially effective for low SNR speech signals. The best setting has achieved a recognition accuracy of 88.2% at -5dB SNR on subway noise task when oracle data mask is used.

Keywords: robust speech recognition, missing feature theory, vector autoregressive model.

1 Introduction

The speech signal is a slowly time-varying signal. The slow time-varying nature is reflected by highly correlated spectral frames, in other words, speech frames are highly predictive. However, current modeling of speech spectral vectors, such as hidden Markov model (HMM) and Gaussian mixture model (GMM) [1][2] usually assume that the spectral vectors of neighboring frames are independent in order to achieve mathematical simplicity. The HMM framework captures the relationship between neighboring frames weakly using the transition probabilities, while GMM completely discards this relationship. It is believed that if the inter-frame statistics are captured and harnessed properly, the performance of both spectral vector reconstruction and speech recognition will be improved.

Noise robustness is an important issue for the implementation of automatical speech recognition (ASR) systems in the real world. Many algorithms have been proposed to improve the recognition accuracies under noisy environments by enhancing the input signals or features. E.g. Wiener filtering, spectral subtraction, cepstral mean normalization, and missing feature theory (MFT) [1] which has become more popular in recent years.

MFT-based techniques usually compensate the corrupted spectral vectors in two steps: the first step is to determine which components of the spectrogram-like time-space representation of the speech (usually in log Mel filterbank domain, hereafter called spectrogram for simplicity) are missing, and the second step is to

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 315–324, 2006.

either reconstruct the missing features for recognition [1,2,3,6] or discard them during the recognition process [3]. Current MFT-based techniques [3,6] usually assume that the neighboring speech frames are independent. E.g., in Cooke's [3] state-based imputation method, the statistics of a log Mel filterbank domain HMM model are used to estimate the missing features. Since the HMM is used, the independence assumption of the spectral features is applied. Although Van Hamme [6] improved the above method by estimating the cepstral coefficients directly from the log Mel filterbank coefficients through a nonnegative least square approach, the independence assumption remained. The above two MFT approaches reconstruct the missing features during the decoding process.

In another MFT approach, instead of reconstructing missing features during decoding, the spectral features are first reconstructed prior to MFCC feature generation [2]. Raj proposed two methods for this approach, the cluster-based and correlation-based reconstruction methods. The cluster-based method assumes the log Mel filterbank coefficient vectors (hereafter called spectral vectors) are from a single independent, identically distributed (IID) process, and uses a GMM to model its distribution. It then reconstructs the missing features using the statistics of the trained GMM and an iterative maximum *a posteriori* (MAP) estimation method. Hence, the cluster based approach does not utilize inter-frame information. In the second method, the correlation-based method, inter-frame statistics are utilized to reconstruct the missing features. The correlation method assumes that the spectral vectors are generated from a single wide-sense stationary multivariate process, and the spectrogram of every utterance is a realization of the process. It first captures the cross-covariances statistics of the spectral features during training and estimates the missing feature using the MAP method. Although inter-frame statistics are utilized, we argue that the full potential of the time information in the spectrogram is not harnessed. One reason is that the speech signal is very dynamic, and a single wide-sense stationary process is insufficient to model the speech spectrogram.

In this paper, we aim to improve the estimation of the missing features giving oracle data mask. We proposed to use the VAR models [4] to model the inter-frame relationship of the speech spectral vectors in a parametric way. The speech spectral vectors are assumed to be from multiple stationary multivariate processes, and for each process, one VAR model is used to capture the inter-frame relationship. Based on these VAR models, two missing feature theory-based feature compensation schemes are proposed. The paper is organized as follows. Section 2 introduces the proposed VAR modeling of spectral vector. Section 3 describes the proposed feature compensation schemes. Section 4 introduces the experimental setup and results. Finally,we conclude in section 5.

2 VAR Modeling of Spectral Vectors

To capture the inter-frame relationship of the speech data in filterbank domain, we propose to model each filterbank channel as an autoregressive (AR) model. The AR model can be used to predict the value of the missing features from either its past neighbors (causal), or future values (anti-causal) in the same filterbank. Furthermore, to capture the inter-filterbank relationship of the speech data, we propose to use the vector-valued extension of the AR model, the VAR model, to model the speech data in filterbank domain. This is because the VAR model utilizes the inter-filterbank relationship to produce more accurate prediction. In the following paragraphs, we will introduce the details of the VAR model for speech data and the Least Square solution for the model parameters.

Let $\mathbf{s}(n)$ be the n^{th} frame of the speech signal, and $\mathbf{x}(n) \in R^D$ is its corresponding log Mel domain spectral vector, where D is the number of Mel windows. We use the VAR models to capture the inter-frame relationship between the spectral vectors such that the current frame feature can be predicted by its past (forward prediction) or future vectors (backward prediction) as illustrated in Fig. 1. The elements of $\mathbf{x}(n)$ are predicted as a linear combination of all the elements of either the past vectors or the future vectors plus a prediction error that is white noise. The mathematical derivation of forward prediction and backward prediction are similar, hence only the derivation of the forward prediction model is discussed. The j^{th} element of $\mathbf{x}(n)$ can be predicted as [4].

$$\begin{aligned} x_j(n) &= f(\mathbf{x}(n-1), \mathbf{x}(n-2), ..., \mathbf{x}(n-P)) \\ &= -\sum_{i=1}^{P} \mathbf{w}_{i,j}^T \mathbf{x}(n-i) + u_j(n) \end{aligned} \tag{1}$$

where $\mathbf{w}_{i,j}$, for $i = 1, ..., P$ are the $D \times 1$ weight vectors, $u_j(n)$ is a white noise or innovation process, P is the order of the model, and $(\cdot)^T$ denotes the matrix transpose. Let $\mathrm{W}_i = [\mathbf{w}_{i,1}, ..., \mathbf{w}_{i,D}]$, $\mathbf{x}(n) = [x_1(n), ..., x_D(n)]^T$ and $\mathbf{u}(n) = [u_1(n), ..., u_D(n)]^T$, where $[\cdot]$ denotes the matrix or vector concatenation, Eq (1) can be rewritten as:

$$\mathbf{x}(n) = -\sum_{i=1}^{P} \mathrm{W}_i^T \mathbf{x}(n-i) + \mathbf{u}(n) \tag{2}$$

where W_i is the $D \times D$ weight matrix for the i^{th} order, $\mathbf{u}(n)$ is the D dimensional white noise.

Here we describe how to find the Least Square solution of W_i [4]. Let $\mathbf{x}(n-i)$ for $i = 1, ..., P$ be concatenated to form a super vector $\mathbf{o}(n)$, i.e. $\mathbf{o}(n) = [\mathbf{x}(n-1)^T, ..., \mathbf{x}(n-P)^T]^T$, and let W_i be concatenated to form $\mathrm{B} = [-\mathrm{W}_1^T, ..., -\mathrm{W}_P^T]$. Then Eq (2) can be rewritten as

$$\mathbf{x}(n) = \mathrm{B}\mathbf{o}(n) + \mathbf{u}(n) \tag{3}$$

To estimate the weight matrix B, we form training samples $\mathbf{r}(n)$ as the concatenation of the desired vector $\mathbf{x}(n)$ and the super input vector $\mathbf{o}(n)$, i.e., $\mathbf{r}(n) = [\mathbf{x}(n)^T, \mathbf{o}(n)^T]^T$. Suppose we have a collection of M training samples, denote the input vector $\mathbf{o}(n)$ of all the samples as $\mathrm{O} = [\mathbf{o}(1), ..., \mathbf{o}(M)]$, and the corresponding desired vectors $\mathbf{x}(n)$ as $\mathrm{X} = [\mathbf{x}(1), ..., \mathbf{x}(M)]$. The Least Square solution can be found by the following equation:

$$\hat{\mathrm{B}} = \mathrm{X}\mathrm{O}^T(\mathrm{O}\mathrm{O}^T)^{-1} = \mathrm{X}\mathrm{O}^{+} \tag{4}$$

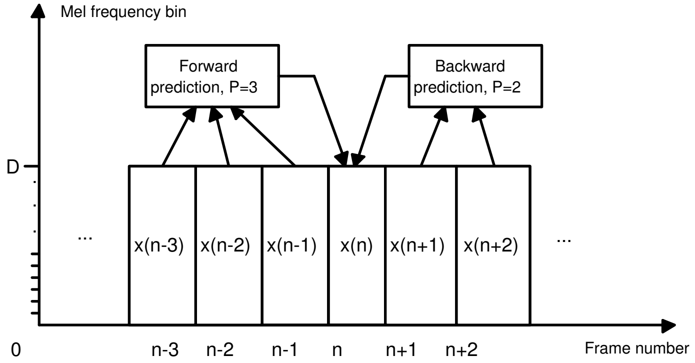


Fig. 1. Illustration of forward and backward prediction

where $\hat{\mathrm{B}}$ is the estimate of B and O^+ is the pseudoinverse of O. The weight matrix $\hat{\mathrm{B}}$ is used to predict the spectral vectors during reconstruction phase, using the formula

$$\hat{\mathbf{x}}(n) = \hat{\mathrm{B}}\mathbf{o}(n) \tag{5}$$

where $\hat{\mathbf{x}}(n)$ is the estimate of $\mathbf{x}(n)$.

Given a collection of training samples, we have described how to construct a VAR model. However, we know that speech signal is not a stationary process. We can not expect to use one VAR model for spectral feature prediction in many different phonetic contexts. Note that speech is composed of a finite number of phonemes. Studies have shown that speech signals of the same phoneme share similar spectral pattern. One solution to vector autoregressive modeling for a ASR task is to have multiple VAR models according to the short-term spectral characteristics. Each VAR is modeled by a collection of homogeneous spectral training samples.

3 Feature Compensation Schemes

This section describes the detailed implementations of the proposed VAR based missing feature reconstruction schemes. We propose two schemes, one uses clean training data, and the other uses noisy training data corrupted by white noise. We present details only for the forward prediction model as the procedure for backward prediction model are similar except the direction of prediction.

3.1 Scheme I: Training on Clean Speech Data

Training. The objective of the training process is to cluster the training samples and estimate the parameters of the VAR models for every cluster. The procedure for clean VAR model training is summarized as follows (see Fig. 2.A):

1. Collect all the training samples $\mathbf{r}(n) = [\mathbf{x}(n)^T, \mathbf{o}(n)^T]^T$ where n denotes the n^{th} frame in the the training set.
2. Use K-means algorithm to cluster all $\mathbf{r}(n)$ into K clusters, called spectral clusters. Estimate the mean vector and covariance matrix of the Gaussian models $\mathcal{N}(\mathbf{r}, \mu_k, \Sigma_k)$ for each cluster $k = 1, ..., K$.
3. Label each sample with a cluster id $k = 1, ..., K$. For each cluster, collect all the input vectors $\mathbf{o}(n)$ and their corresponding desired vectors $\mathbf{x}(n)$, then estimate the weight matrix B_k of cluster k using Eq (4).

Testing. The procedure of estimating missing features is as follows (Fig. 2B):

1. For each noisy vector $\mathbf{x}(n)$, identify the set of reliable and unreliable features. We use the oracle mask to do so in this paper. For each of the unreliable features, go through step 2-4;
2. Estimate the spectral vector $\mathbf{x}(n)$ using the VAR models, we first form a vector $\mathbf{r}(n) = [\mathbf{x}(n)^T, \hat{\mathbf{x}}(n-1)^T, ..., \hat{\mathbf{x}}(n-P)^T]^T$, where the first vector is the noisy vector $\mathbf{x}(n)$ and $\{\hat{\mathbf{x}}(n-j)\}_{j=1}^{P}$ are the reconstructed past vectors.
3. Find the *a posteriori* probability for every Gaussian model $\mathcal{N}(\mathbf{r}, \mu_k, \Sigma_k)$ for $k = 1, ..., K$ given vector $\mathbf{r}(n)$ in the same manner as in the cluster-based method reported in [2].

$$p(\mathbf{r}(n); k) = \mathcal{N}(\mathbf{r}(n), \mu_k, \Sigma_k), \quad k = 1, ..., K \tag{6}$$

$$p(k|\mathbf{r}(n)) = p(\mathbf{r}(n); k) / \sum_{l=1}^{C} p(\mathbf{r}(n); l) \tag{7}$$

   where $p(\mathbf{r}(n); k)$ is the likelihood of $\mathbf{r}(n)$ on the k^{th} Gaussian model and $p(k|\mathbf{r}(n))$ is the *a posteriori* probability.
4. Form super input vector $\mathbf{o}(n) = [\hat{\mathbf{x}}(n-1)^T, ..., \hat{\mathbf{x}}(n-P)^T]^T$. Find the model-dependent prediction $\tilde{\mathbf{x}}_k(n)$ of $\mathbf{x}(n)$ using

$$\tilde{\mathbf{x}}_k(n) = \mathrm{B}_k \mathbf{o}(n), \quad k = 1, ..., K \tag{8}$$

5. Reconstruct the feature vector $\mathbf{x}(n)$ by

$$\hat{x}_j(n) = \begin{cases} min\{x_j(n)\ , \sum_{k=1}^{K} p(k|\mathbf{r}(n))\tilde{x}_{kj}(n)\ \}, \\ \quad\quad \text{for unreliable features;} \\ x_j(n), \text{ for reliable features.} \end{cases} \tag{9}$$

   where the reliable features keeps original values while the unreliable features are replaced by the weighted sum bounded to the corresponding noisy values, and $\tilde{x}_{kj}(n)$ is the j^{th} element of $\tilde{\mathbf{x}}_k(n)$. Instead of using a hard decision in model selection, we form the estimate for a missing feature using a linear combination of its estimates from all models, where the weights are the *a posteriori* probabilities $p(k|\mathbf{r}(n))$ of models.

In the experiments, we found that good performance is achieved by using both the forward and backward model to reconstruct the final vector by simply

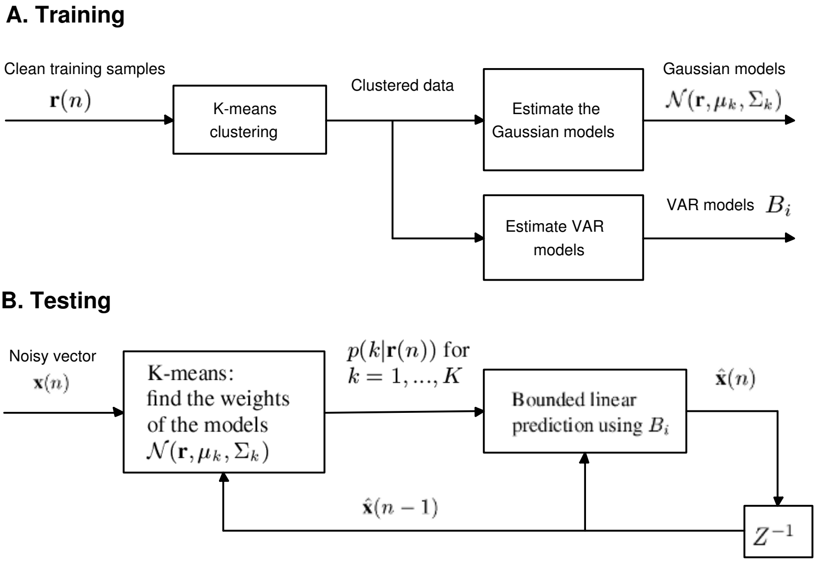

A. Training
Clean training samples
$\mathbf{r}(n)$
K-means clustering
Clustered data
Estimate the Gaussian models
Gaussian models
$\mathcal{N}(\mathbf{r}, \mu_k, \Sigma_k)$
Estimate VAR models
VAR models B_i
B. Testing
Noisy vector
$\mathbf{x}(n)$
K-means: find the weights of the models $\mathcal{N}(\mathbf{r}, \mu_k, \Sigma_k)$
$p(k|\mathbf{r}(n))$ for $k = 1, ..., K$
Bounded linear prediction using B_i
$\hat{\mathbf{x}}(n)$
$\hat{\mathbf{x}}(n-1)$
Z^{-1}

Fig. 2. Scheme I: Training on clean data and testing on noisy data

averaging the two predictions. In this way, both information from past vectors and future vectors are fully utilized.

Using clean data to train the spectral clusters and their corresponding VAR models gives rise to two types of mismatches. First, the spectral clusters use probabilistic mixture to model the distribution of the clean training data. As a result, the derived spectral clusters do not describe well distribution of noisy speech data in run-time, leading to inaccurate estimate of cluster *a posteriori* probability $p(k|\mathbf{r}(n))$. The estimate of $p(k|\mathbf{r}(n))$ has direct impact on the quality of reconstructed missing feature. Second, VAR model relies on a sequence of spectral frames to predict a new frame. If the VAR model is trained on clean data, by taking corrupted data in run-time, the VAR model's performance would be unexpected. To address the two mismatches in Scheme I. We propose Scheme II that trains the system using data corrupted by white noise.

3.2 Scheme II: Training on Noisy Speech Data

In this scheme, noisy data are used to train the system as illustrated in Figure 3A. Two approaches are studied, the first uses the noisy data directly to train the system, while the second approach preprocesses the data prior to system training. The training procedure is similar to that of the clean training with two differences. First, the noisy data are used for the spectral clustering; second, the weight matrices B_i are trained by minimizing the prediction error using noisy input vectors to predict the clean desired vector. When reconstructing missing features, the calculation of the *a posteriori* probabilities of VAR models is based

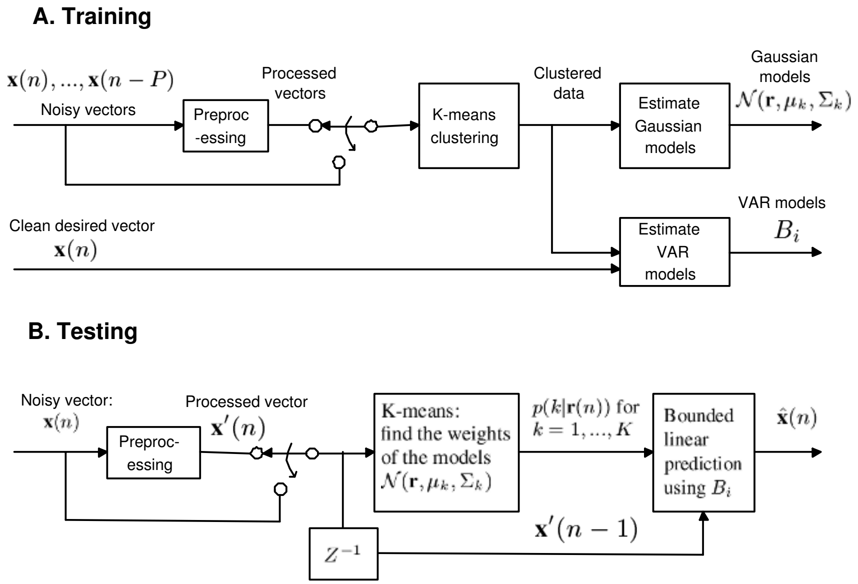


Fig. 3. Scheme II: Training on clean and noisy data, testing on noisy data, with and without preprocessing

on the noisy spectral vectors. The prediction is also based on the noisy spectral vectors of neighboring frames. Therefore, the accuracy of the calculation of the *a posteriori* probability is improved and the mismatch in feature prediction is minimized.

Although the noisy training scheme reduces the mismatches that exists for clean training scheme, there are some inherent technical constraints for this scheme. First, the statistics of the noisy signal changes with signal to noise ratio (SNR). Hence models trained on data of one SNR level are not adequate to reconstruct the noisy speech at another SNR level. Second, for very poor SNR cases ($<$ 5dB), we found that the accuracy of the *a posteriori* probability is low which results in poor reconstruction of features.

To alleviate these problems, the preprocessing module is incorporated into the noisy training and testing processes when the switch chooses preprocessed data $\mathbf{x}'(n)$ (see Fig. 3A&B). The objective of the preprocessing module is to condition the noisy speech signal prior to training or testing. Specifically, it aims at reducing the mismatch caused by SNR difference and producing more reliable features. The latter is important, as we assume that for the VAR prediction to be effective, enough reliable features need to be present in the input vectors. By making more reliable features, the performance of the VAR prediction will be improved. Many feature compensation and speech enhancement methods may be used for preprocessing, such as Wiener filter and spectral subtraction [5]. In our experiments, the cluster-based feature compensation method [2] is used for preprocessing.

4 Experimental and Results

4.1 Experiments Setup

The AURORA-2 database [7] was used to evaluate the performance of the proposed feature compensation schemes. The training and testing of the recognition engine follow the scripts provided by the database, except that the c0 is used, rather than log energy. Due to space limits, we only report results on subway noise of test set A and restaurant noise of test set B. As our objective is to examine the performance of the proposed schemes to reconstruct the missing features, we used the oracle binary data mask for our experiments. The optimal SNR threshold is found to be -5dB by experiments [1].

For our two proposed schemes, our experimental results showed that increasing number of clusters improves performance and $C = 50$ cluster is sufficient to model the different classes of speech segments for the AURORA-2 database. We also found $P = 3$ to be a suitable VAR order for the experiments. Hence, these two parameters are used throughout in our experiments as discussed in the following sections.

4.2 Experimental Results Using Oracle Data Mask

The following six results were obtained for the AURORA-2 Test Set A subway noise as illustrated in Fig 4.

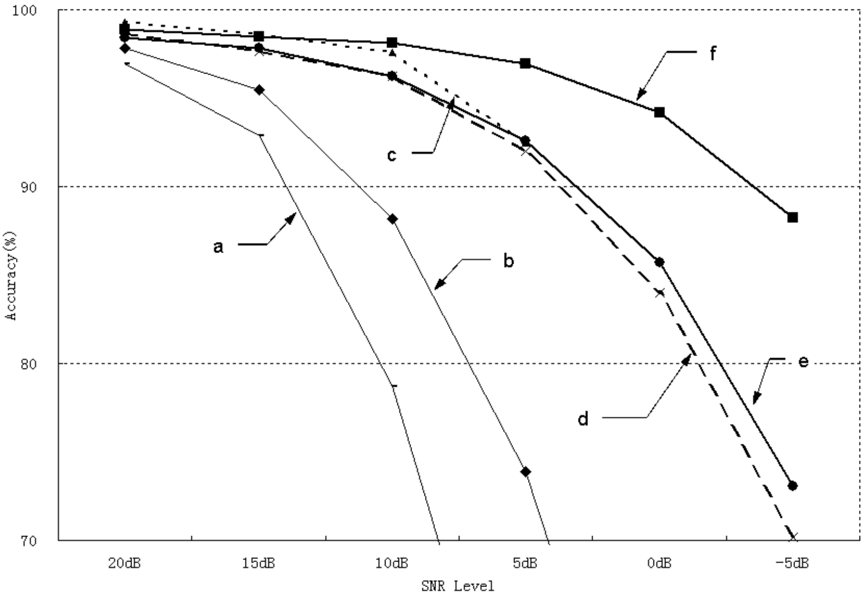


Fig. 4. Recognition results on subway noise of Test Set A

a) AURORA-2 baseline model using clean training.
b) Proposed scheme II without preprocessing.
c) Van Hamme's reported results from [6] using oracle mask.
d) Raj's [1,2] cluster based MFT method with 20 clusters using oracle mask.
e) Proposed scheme I.
f) Proposed scheme II with preprocessing.

The results showed that our proposed noisy training Scheme II with preprocessing (line-f) gives the best performance with absolute accuracy 88.2% at SNR = -5dB. Compare this result to line-b (noisy training scheme II without preprocessing), the dramatic improvement highlights the importance of the preprocessing steps that conditions the input vectors for the VAR models. Note that the preprocessing step applied for line-f is actually that of Raj's clustering [1,2] as in line-d. As line-f is significantly better than line-d, this shows that the VAR model further utilizes the inter-frame relationship to improve the reconstruction performance. It implies that the VAR model and Raj's clustering uses complementary information, i.e. the the cluster-based preprocessing module only exploit the intra-frame relationship while the VAR exploits the inter-frame ones. Our experimental results have showed that better result are obtained when these two kinds of information are used together properly. The results of our proposed clean training Scheme I (line-e) indicates the performance of the VAR-alone scheme. It produces similar performance as Raj's cluster-based method (line-d).

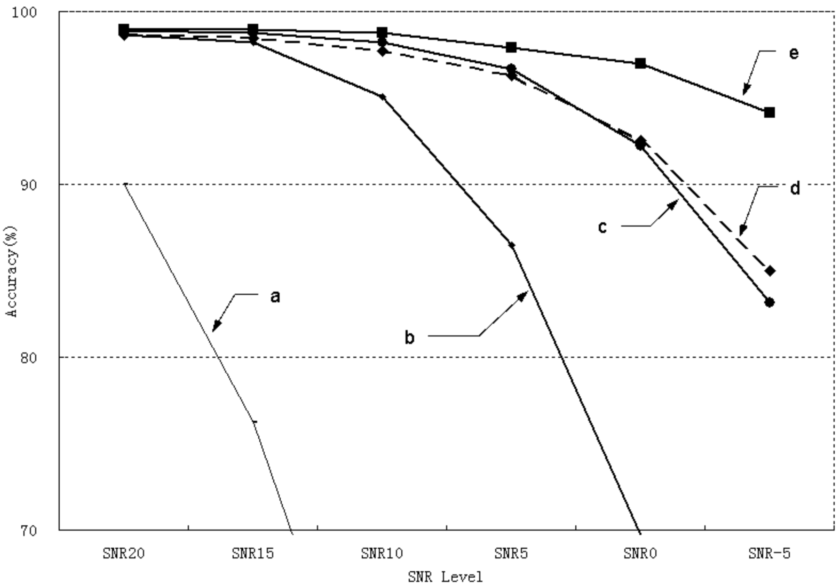


Fig. 5. Recognition results on restaurant noise of Test Set B

This shows that the inter-frame information is as effective as the intra-frame information on the task of missing feature reconstruction.

The results for the restaurant noise of Test Set B shows similar relative performance as that of subway noise (Fig. 5). Here line-a is the AURORA-2 baseline; line-b is for Scheme II without preprocessing; line-c is for Scheme I; line-d is Raj's cluster based MFT reconstruction; and line-e is for Scheme II with preprocessing.

5 Conclusion

In this paper, we proposed two novel feature compensation schemes using the Vector Autoregressive modeling of spectral vectors. The VAR models are trained on both clean and noisy data respectively. It is found that the models trained on noisy data with the use of preprocessing module provides the best recognition accuracies. The improvement can be credited to the use of both the interframe and intraframe information during feature compensation.

Future research may be conducted to improve the prediction accuracy using nonlinear prediction and use other types of preprocessing techniques. In addition, we will also examine the use of GMM for the spectral clustering. Although our experimental results showed its effectiveness, it is an empirical method and better method for clustering spectral vectors may be used to improve the overall performance. The estimation of VAR model parameters can also be more robust by using advanced model identification methods [4].

References

1. B. Raj, R.M. Stern, "Missing-feature approaches in speech reconitoin", *Signal Processing Magazine*, vol. 22, no 5, pp. 101-116, Sep. 2005
2. B. Raj, M.L. Seltzer, R.M. Stern, " Reconstruction of missing features for robust speech recognition", *Speech Communication*, vol. 43, no. 4, pp. 275-296, Sep. 2004
3. M. Cooke, P. Green, L. Josifovski, Vizinho, A., "Robust automatic speech recognition with missing and uncertain acoustic data", *Speech Communication*, vol. 34, pp. 267-285, 2001
4. H. Lütkepohl, "Introduction to multiple time series analysis", 2nd Ed., Springer-Verlag, 1993
5. J.S. Lim and A.V. Oppenheim, "Enhancement and bandwidth compression of noisy speech", *Proceeding of the IEEE*, vol.67, no.12, pp.1586-1604, Dec. 1979
6. H. Van hamme, "Robust speech recognition using cepstral domain missing data techniques and noisy mask", *ICASSP* 2004
7. D. Pearce, H.-G. Hirsch, "The AURORA experimental framework for the performance evaluation of speech recogntion systems under noisy conditions", *ICSLP*, Beijing, China 2000, pp. 29-32

Auditory Contrast Spectrum for Robust Speech Recognition

Xugang Lu and Jianwu Dang

School of Information Science, Japan Advanced Institute of Science and Technology
1-1, Asahidai, Nomi, Ishikawa 923-1211, Japan

Abstract. Traditional speech representations are based on power spectrum which is obtained by energy integration from many frequency bands. Such representations are sensitive to noise since noise energy distributed in a wide frequency band may deteriorate speech representations. Inspired by the contrast sensitive mechanism in auditory neural processing, in this paper, we propose an auditory contrast spectrum extraction algorithm which is a relative representation of auditory temporal and frequency spectrum. In this algorithm, speech is first processed using a temporal contrast processing which enhances speech temporal modulation envelopes in each auditory filter band and suppresses steady low contrast envelopes. The temporal contrast enhanced speech is then integrated to form speech spectrum which is named as temporal contrast spectrum. The temporal contrast spectrum is then analyzed in spectral scale spaces. Since speech and noise spectral profiles are different, we apply a lateral inhibition function to choose a spectral profile subspace in which noise component is reduced more while speech component is not deteriorated. We project the temporal contrast spectrum to the optimal scale space in which cepstral feature is extracted. We apply this cepstral feature for robust speech recognition experiments on AURORA-2J corpus. The recognition results show that there is 61.12% improvement of relative performance for clean training and 27.45% improvement of relative performance for multi-condition training.

Keywords: Auditory model, adaptation, contrast spectrum, speech recognition.

1 Introduction

Traditional speech representations are based on power spectrum, such as MFCC, PLP, etc.[1], before doing nonlinear compression and de-correlation, speech power or amplitude spectrum is usually resulted by integration from some frequency channels using filter bands. Since all frequency components are taken into consideration, these kinds of representations are sensitive to noise distortion because noise energy distributed in some frequency bands may deteriorate speech representations. For speech recognition, we only need the speech information which is kept from noise. For extracting such information, we must investigate two aspects; one aspect is from the difference of statistic properties between noise and speech, another aspect is from auditory neural processing, i.e. how does the auditory system deal with the difference.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 325 – 334, 2006.

Usually, the statistic properties of speech are changed more rapidly than those of most of the noises caused by environments, even for a non-stationary noise. So the dynamic changes of statistic characters can be used to discriminate speech from noise. In addition, speech is produced by speech organs with certain resonance properties, thus the spectral profiles show regular structures, such as harmonic structure (for voiced speech), formant structure, etc.[2]. These spectral profiles should be distinguished from those of other acoustic sounds produced by physical entities, such as car, train, etc. Enhancing the dynamic changes and regular spectral structures of speech should be helpful for robust speech representations. From auditory neural processing aspect, the extraction of the dynamic changes and enhancing of speech spectral structures mechanisms do exist [3][4][5][7]. These mechanisms can be regarded as temporal and frequency contrast processing of speech. However, there is no a unifying model to deal with the contrast processing for robust speech representations. Some models treat the contrast processing implicitly even without considering the different characteristics of speech and noise. In this paper, we propose an algorithm based on the auditory contrast processing mechanisms which realizes the temporal and frequency contrast processing of acoustic signals explicitly. During the design of the algorithm, we take the differences of speech and noise characteristics into consideration, and directly adapt the algorithm for robust speech feature extraction. The left of this paper is organized as follows. In section 2, we briefly introduce the temporal and frequency contrast processing in auditory system, and design an algorithm to realize the contrast processing functions. In section 3, we adapt the proposed algorithm for robust speech feature extraction and test the robustness on speech recognition task. In last section, we give some discussions and conclusions.

2 Auditory Contrast Processing

Contrast sensitivity mechanism is common in biology neural processing for different stimulations. In computational auditory model, the contrast processing function for acoustic stimulations also has its neural basis [5][6][7]. Most of the computational auditory models for speech feature extraction usually use a static input/output (I/O) or input/firing (I-F) response curve to describe the relationship between the stimulation and the response of an auditory neuron. The feature extracted based on this static response curve represent the stimulation intensity. However, auditory neurons can adapt to a temporal varying stimulation continuously which shows context dependent responses. In this adaptation response, neurons distribute more firing sparks to the high temporal contrast parts of stimulations while allocate only a few firing spikes for steady stimulations or low contrast parts of stimulations. It is proved that this kind of adaptation is important for improving the neurons' information transfer efficiency, reducing the information redundancy [6]. In our application, considering the different statistic changes of speech and noise, we want to use this mechanism to enhance the high temporal contrast parts of acoustic signals for speech feature representation, thus the representation is a temporal contrast representation rather than an absolute intensity representation.

2.1 Auditory Temporal Contrast Processing

For combining the temporal contrast processing of neurons with our auditory speech feature extraction model, in this paper, we choose an adaptive compression loop model. This model was originally used to quantify psychoacoustic experiments, and later was also applied for robust speech recognition [7]. The model is composed of five adaptation loops as shown in Fig.1.

In Fig.1, the $a(t)$ is the input acoustic signal (from one auditory neural fiber), $y(t)$ is the output of the adaptation model. $R_i, C_i, i=1,2,...,5$ are the resistor and capacitor with a time constant $\tau_i, i=1,2,...,5$ respectively. This model can detect the temporal dynamic changes of acoustic stimulations while suppress the slow and high varying parts (based on the perception of speech intelligibility, the temporal modulation frequency below 1Hz is regarded as slow dynamic changes while the temporal frequency above 20Hz is regarded as high varying parts).

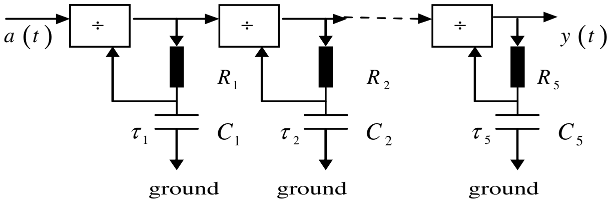


Fig. 1. Nonlinear adaptation model (modified based on [7])

The processing effect of the adaptation model is shown in Fig.2, the upper panel is the original temporal envelope and the lower panel for contrast envelope processed by one auditory frequency channel with center frequency 1k Hz. One can see from Fig.2 that the temporal contrast processing detects the relative changes of speech stimulations rather than detecting the intensity of acoustic stimulations. Since usually noise signals are more stationary compared to speech signals, the local contrast processing can serve as a suppressor of noise signals.

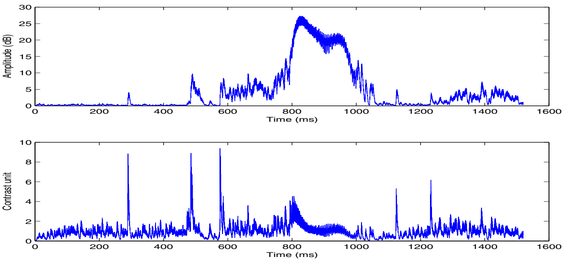


Fig. 2. Temporal contrast processing by the temporal adaptation model. Original temporal envelope (top), temporal contrast envelope (bottom).

2.2 Auditory Frequency Contrast Processing

In auditory processing, the topological structure of the frequency is kept well while being transferred to auditory cortex [3][5]. There are some interactions between neighboring frequency channels. One of the most important interactions is the lateral inhibition mechanism. The function of the lateral inhibition is to enhance spectral contrast [5]. Most of the applications of the lateral inhibition models do not consider the difference of spectral profiles between speech and noise signals. In our analysis, we apply the lateral inhibition mechanism by considering the difference of spectral profiles between speech and noise signals.

For a simple analytic analysis purpose, in this paper, we choose a normalized second order derivative of Gaussian function which fulfils the lateral inhibition effect well. The Gaussian function with a variance parameter c and its normalized form are represented as:

$$g\left(x,c\right)=\frac{1}{c\sqrt{2\pi}}\exp\left(-\frac{x^2}{2c^2}\right), \; f\left(x,c\right)=\frac{g\left(x,c\right)}{g\left(0,c\right)}=\exp\left(-\frac{x^2}{2c^2}\right) \tag{1}$$

The lateral inhibition function which can be regarded as a scale function is gotten as:

$$\varphi\left(x,c\right)=-\frac{\partial^2 f\left(x,c\right)}{\partial x^2}=\frac{1}{c^2}\left(1-\frac{x^2}{c^2}\right)\exp\left(-\frac{1}{2}\frac{x^2}{c^2}\right) \tag{2}$$

The function is applied on a spectral profile as:

$$y\left(\omega,c\right)=x\left(\omega\right)*\varphi\left(\omega,c\right) \tag{3}$$

where * is the convolution operator, $x(\omega)$ is the original auditory spectral profile, and $y(\omega,c)$ is the auditory scale spectral profile analyzed using a scale function $\varphi(\omega,c)$ with scale parameter c .

The scale function (2) with two scale parameters and corresponding frequency response (spatial frequency) is shown in the left and right panels of Fig.3 respectively. From Fig.3, one can see that if a large scale parameter of the scale function is used for spectral profile processing, the spectral peaks with wide bandwidths in spectral structure will be emphasized, while for a small scale parameter, spectral peaks with narrow

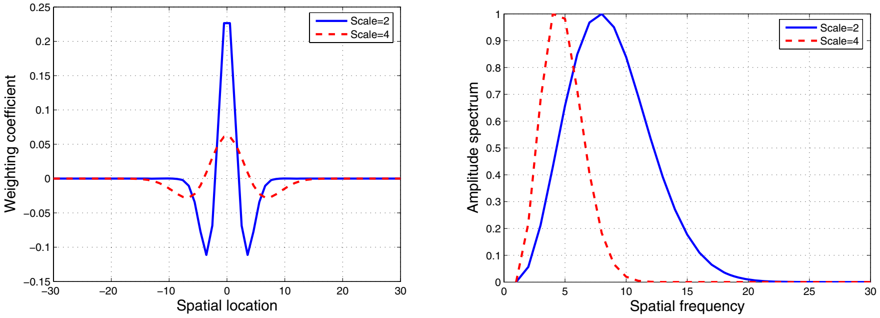


Fig. 3. Scale analysis, impulse response of the scale functions (left) and normalized spatial frequency response (right) with scale parameters 2(dashed curve) and 4(solid curve)

bandwidths will be emphasized. Thus the band-pass filtering characteristic of the scale function makes the processing enhance the spectral contrast in some spatial scales.

For a noisy speech spectral profile, the noise effect is different in each spectral scale space. For measuring the noise effect on speech spectral profile in different scale spaces, we analyze the noise distortion on auditory contrast spectral profiles using speech sentences from AURORA-2J speech corpus [9]. We use a normalized L2 distance between clean and noisy spectral profiles in scale spaces to measure the noise distortion on speech (the normalized L2 distance is calculated as the Euclidian distance of the two compared spectral profiles which are normalized by maximum spectral profile values sentence by sentence). Fig. 4 shows the normalized L2 distance between a clean and noisy speech spectral profiles in different scale spaces for subway and restaurant noises under different SNR conditions. From Fig.4 one can see that the noise distortion increases with the decreasing of SNR levels. The distortion gets a minimum when the scale index approximating 3 in both noise types for almost all SNR conditions. That is to say, the optimal scale space seems to depend more on speech spectral profiles rather than on noise types. For finding a good spectral scale space, we define an objective function for one spectral profile which consists of two parts, one part is related with noise reduction, and another part is related with speech distortion as shown in formula (4).

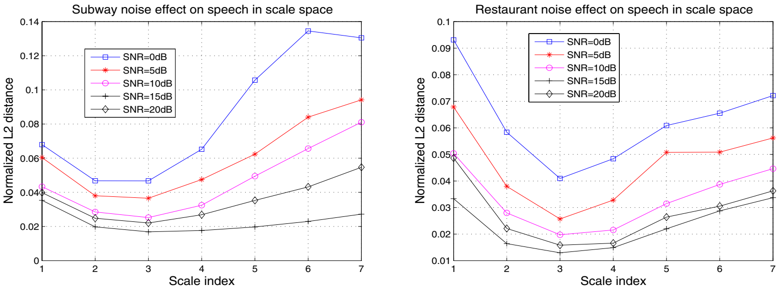


Fig. 4. Normalized L2 distance between clean and noisy spectral profiles in different scale spaces (subway and restaurant noise)

$$e(\omega,c)=\underbrace{\left|y(\omega,c)-y_c(\omega,c)\right|^2}_{noise\quad reduction}+\lambda\underbrace{\left|x(\omega)-y(\omega,c)\right|^2}_{speech\quad distortion} \qquad (4)$$

where $x(\omega)$ is the clean speech spectral profile, $y(\omega,c)$ and $y_n(\omega,c)$ are the clean and noisy speech scale spectral profiles, λ is the weighting coefficient which balances the importance of noise reduction and speech distortion. Considering the different dynamic ranges of noise reduction and speech distortion, we normalize noise reduction and speech distortion components to the same ranges sentence by sentence. The weighting coefficient λ is chosen as 1 in this paper to show the equal importance of noise reduction and speech distortion. By minimizing the objective function $e(\omega,c)$, we can get an optimal scale parameter for the noisy speech spectral profile $x_n(\omega)$:

$$c^{*} = \arg\min_{c}\left(e\left(\omega, c\right)\right) \qquad (5)$$

In real application, the scale parameter should be different for different noise under different SNR levels. But currently, we use an average scale parameter for all. For an initial experiment, we randomly choose ten clean speech sentences, and corresponding noisy speech sentences from test A, test B and test C at the SNR 0dB, 5dB, 10dB, 15dB and 20dB from AURORA-2J [9]. We added the objective function (5) for all the speech sentences under all SNR conditions to get a total objective function. Based on the total objective function, we got an average optimal scale parameter value as 3.42. Comparing the optimization result with the normalized L2 distances in scale space showed in Fig.4 , it is reasonable to assume that the scale parameter value 3.42 can be a good choice for the AURORA-2J speech data for all noisy conditions (although not an optimal value for each noise type respectively). We show an example of the auditory contrast spectrum in Fig.5 using the optimized scale parameter. From the figure, one can see that the noise effect is reduced more in auditory contrast spectrum representation than that of the auditory power spectrum representation.

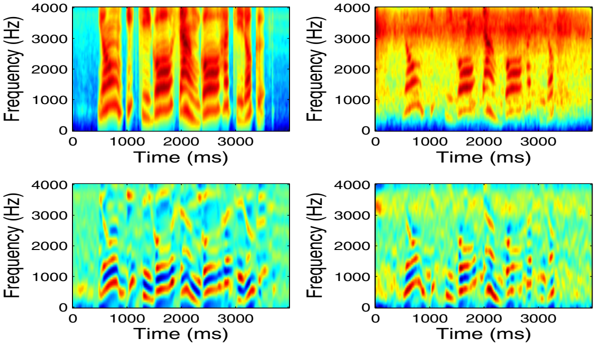


Fig. 5. Comparison of auditory power spectrum (upper two panels) and auditory contrast spectrum (lower two panels) for clean speech (left two panels) and noisy speech (right two panels) (SNR=5dB, subway noise)

3 Speech Recognition Based on Auditory Contrast Spectrum

We test the performance of the contrast spectrum on robust speech recognition experiments. The AURORA-2J corpus is adopted. The feature vector and Hidden Markov Model (HMM) configurations are the same as used for standard comparisons [9].

3.1 Auditory Contrast Spectral Feature Extraction

For auditory contrast spectral feature extraction, we add two additional processing blocks to extract the temporal and frequency contrast feature of speech in a common auditory processing flowchart. The processing blocks are shown in Fig.6. The two dashed rectangle blocks are the added contrast processing blocks. The features extracted with and without the contrast processing blocks are denoted as: (a) AFC: Auditory frequency power spectrum based cepstral feature extracted without the two contrast processing blocks. (b) FC-AFC: Frequency contrast AFC extracted with the frequency contrast processing block only. (c) TC-AFC: Temporal contrast AFC extracted with the temporal contrast processing block only. (d) FC-TC-AFC: Frequency and temporal contrast AFC extracted with both the two contrast processing blocks.

The procedures for extracting FC-TC-AFC are described as: (1) Observed acoustic signals are band-pass filtered using N=60 gammatone band-pass filters which fulfill the function of basilar membrane. Then the output of each band-pass filter is half-wave rectified and low-pass filtered which serve as the functions of inner hair cells and auditory neural fibers. (2) The nonlinear adaptation model in Fig.1 is used for temporal contrast processing in each frequency band followed by a temporal integration which integrates 20ms of the temporal contrast signal to produce the temporal contrast spectrum (with 10ms frame shift). (c) The temporal contrast spectrum from all frequency channels is processed using a spectral profile scale analysis to extract the contrast spectrum followed by a DCT to get the cepstral feature for HMM.

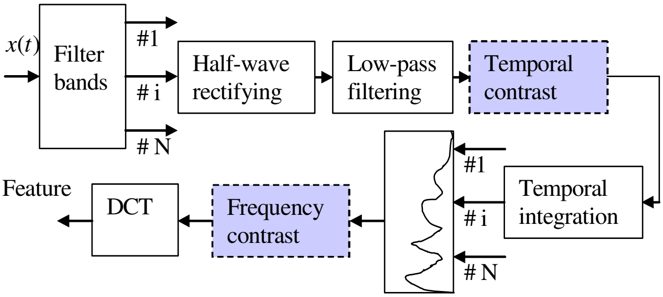


Fig. 6. Auditory contrast spectrum feature extraction

Based on the four feature representations, the recognition performance is shown in Fig.7 for test A, test B, test C and the overall performance. In Fig.7, the recognition performance of AFC is a little higher than that of MFCC which is used as a baseline for comparison in our experiments. From the figure, one can see that the temporal and frequency contrast processing do improve the robust performance of the speech recognition.

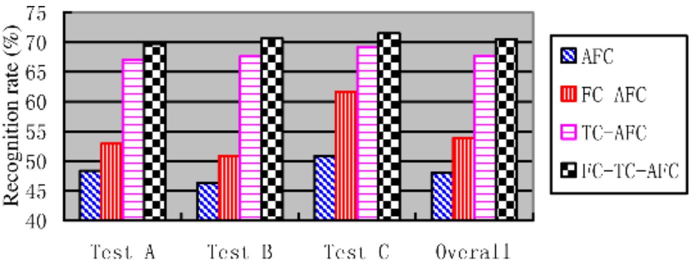


Fig. 7. Speech recognition rate for four types of representations under clean training condition

3.2 Comparison with Other Noise Reduction and Feature Extraction Methods

We compare the proposed contrast spectra based feature (FC-TC-AFC) with two robust features extracted using minimum statistic based spectral subtraction (SpecSub) [10] and optimally modified least square amplitude spectral estimation (OM_LSA) [11]. The recognition results are shown in Fig.8. Seeing from Fig.8, one can see that the auditory spectrum with contrast processing (FC-TC-AFC) representation is the most robust among the compared representations.

3.3 Normalized Auditory Contrast Spectral Representation

Usually the mean and variance normalization on the feature vectors make the representation more robust to noise. The mean normalization is done using:

$$\overline{v}(i,t) = v(i,t) - E_t\left[v(i,t)\right], i = 1,2,...,m \ ; \ t = 1,2,...,n \tag{6}$$

where $v(i,t)$ is the original cepstral coefficient, $\overline{v}(i,t)$ is the mean normalized cepstral coefficient, i is the cepstral order index, t is time frame index. $E_t(.)$ is the expectation operator on time dimension.

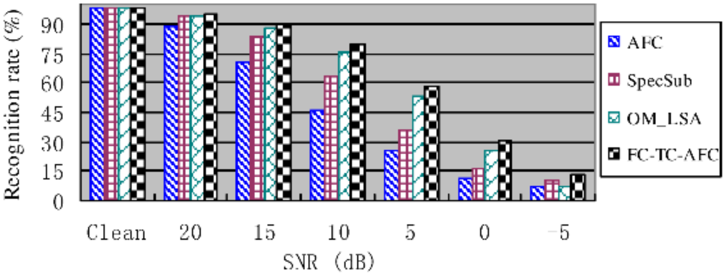


Fig. 8. Speech recognition performance under clean training condition for the four compared representations

The variance normalization is done using:

$$\tilde{v}(i,t) = \frac{\overline{v}(i,t)}{\sqrt{var_t\left(\overline{v}(i,t)\right)}}, \ i = 1,2,...,m \ ; \ t = 1,2,...,n \tag{7}$$

where $\tilde{v}(i,t)$ is the variance normalized cepstral coefficient, $var_t(.)$ is the variance operation on time dimension. In this paper, the mean and variance normalization is done based on the mean and variance of cepstral coefficient for each sentence. For the normalized representations, a prefix N- is used to each original representation symbol, i.e., N-AFC, N-SpecSub, N-OM-LSA, N-FC-TC-AFC. Based on the normalized representations, the recognition performance for clean training is shown in Fig.9. Table 2 shows the relative recognition performance. One can see from figure 9 and table 2 that all the normalized representations improve the robust performance. But the improvements are different for different representations, and representations based on auditory model processing get higher improvements than the other two representations do. Among all the compared representations, the performance of our proposed N-FC-TC-AFC representation is the best. For multi-conditional training [9], the relative performances of normalized representations are shown in table3. From tables 2 and 3, one can see that although the N-SpecSub and N-OM-LSA improve the robust performance in clean training condition, they degrade the robust performance in multi-training condition. But the N-AFC and N-FC-TC-AFC both improve the robust performance in the two training conditions.

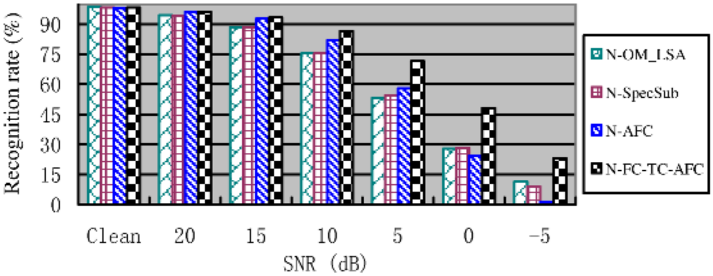


Fig. 9. Recognition performance under clean training condition for normalized representations

Table 2. Relative performance for clean training condition

	Test A	Test B	Test C	Overall
N-OM-LSA	40.39%	43.63%	32.45%	40.26%
N-SpecSub	41.28%	48.01%	24.31%	40.92%
N-AFC	43.29%	47.19%	45.64%	45.35%
N-FC-TC-AFC	59.58%	62.72%	60.83%	61.12%

Table 3. Relative performance for multi-conditional training

	Test A	Test B	Test C	Overall
N-OM-LSA	-50.99%	15.70%	10.46%	-1.42%
N-SpecSub	-61.20%	21.13%	-16.29%	-6.24%
N-AFC	-20.26%	30.53%	29.40%	18.07%
N-FC-TC-AFC	-7.22%	39.61%	35.22%	27.45%

4 Discussions and Conclusion

In this paper, we proposed an auditory contrast spectrum representation for speech recognition. The representation enhances the temporal and frequency contrast components, which is a relative and acoustic context-dependent representation of speech. The representation does not take the absolute physical speech spectral information as a feature, but takes the relative temporal and frequency contrast information as a feature. Our experiments showed that the adding of the temporal and frequency contrast processing blocks improve the robustness of the speech recognition performance. For normalized contrast spectral representation, it got 61.12% improvement of relative performance for clean training, and 27.45% improvement of relative performance for multi-condition training. However, there are some problems needing to be investigated further. For the selection of the optimal scale parameter for spectral profile processing, we only used a small collection of speech sentences and mixed with some types of noise. We assumed that the optimal scale parameter is depended more on the speech spectral profiles rather than on noise types. This assumption is only kept for averagely flatten noise spectral types, however, may not always be kept for many other noise types. In addition, the scale parameter should be an adaptive value which can be adapted to different speech and noise situations. Another problem is how to choose the balance between noise reduction and speech distortion in a scale space for minimization. In future, we will investigate a more general scale analysis model for taking the discussed problems into consideration.

References

1. L. Rabiner, B.H. Juang: Fundamentals of Speech Recognition. Prentice Hall PTR, 1993.
2. K.N. Stevens: Acoustic phonetics. MIT press, 1998.
3. Steven Greenberg, et al: Speech processing in auditory system. Springer-Verlag New York, 2004.
4. J.O. Pickles: An introduction to the physiology of hearing. London, Academic press inc. Ltd. 1982.
5. S. Shamma: Speech processing in the auditory system, II: Lateral inhibition and the central processing of speech evoked activity in the auditory nerve. J.Acoust.Soc.Am., Vol78, pp. 1622-1632, 1985.
6. A.Fishbach, et al: Auditory edge detection: a neural model for physiological and psychoacoustical responses to amplitude transients. J. Neurophysiol., 85, pp2303-2323, 2001.
7. Dau, T., et al: A quantitative model of the effective signal processing in the auditory system. I. Model structure. Journal of the Acoustical Society of America, vol.99, 3615-3622, 1996.
8. D. G. Stork and H. R. Wilson: Do Gabor functions provide appropriate descriptions of visual cortical receptive fields?. Journal of the Optical Society of America A, vol. 7, no. 8, pp. 1362--1373, 1990
9. Http://sp.shinshu-u.ac.jp/CENSREC/, About AURORA- 2J database.
10. R.Martin: Spectral subtraction based on minimum statistics. Eurospeech'94, pp.1182-1185, 1994.
11. I.Cohen, B. Berdugo: Speech enhancement for non-stationary noise environments. Signal processing, Vol.81, pp.2403-2418, 2001.

Signal Trajectory Based Noise Compensation for Robust Speech Recognition*

Zhi-Jie Yan[1], Jian-Lai Zhou[2], Frank Soong[2], and Ren-Hua Wang[1]

[1] University of Science and Technology of China, Hefei, 230027
[2] Microsoft Research Asia, Beijing, 100080
yanzhijie@ustc.edu, {jlzhou, frankkps}@microsoft.com, rhw@ustc.edu.cn

Abstract. This paper presents a novel signal trajectory based noise compensation algorithm for robust speech recognition. Its performance is evaluated on the Aurora 2 database. The algorithm consists of two processing stages: 1) noise spectrum is estimated using trajectory auto-segmentation and clustering, so that spectral subtraction can be performed to roughly estimate the clean speech trajectories; 2) these trajectories are regenerated using trajectory HMMs, where the constraint between static and dynamic spectral information is imposed to refine the noise subtracted trajectories both in "level" and "shape". Experimental results show that the recognition performance after spectral subtraction is improved with or without trajectory regeneration, but the HMM regenerated trajectories yields the best performance improvement. After spectral subtraction, the average relative error rate reductions of clean and multi-condition training are 23.21% and 5.58%, respectively. And the proposed trajectory regeneration algorithm further improves them to 42.59% and 15.80%.

1 Introduction

The performance of a speech recognizer trained in clean condition degrades dramatically in the presence of noise. This degradation is one of the major problems that still remain unsolved in speech recognition. To deal with the noise robustness problem, varieties of noise compensation techniques have been proposed. Among them, spectral subtraction [1] is one of the most popular methods for noise robust speech recognition.

Spectral subtraction makes an estimate of the noise spectrum, which is then subtracted from the corrupted speech spectrum, to get an evaluation of the underlying "clean" speech. Because the performance of spectral subtraction relies mostly on the estimate of noise, the labeling of speech and non-speech regions for noise tracking and estimate becomes a key problem. While former algorithms usually use a Voice Activity Detector (VAD) to label the gaps of speech as noise, recent researches of spectral subtraction aim to label the non-speech regions

* This work has been done when the first author was a visiting student with Speech Group, Microsoft Research Asia.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 335–345, 2006.

not simply along the time axis (the speech gaps), but also along the frequency axis (the so called "harmonic tunneling") [2, 3]. However, noise labeling and estimation when only one single corrupted channel is available has always been a difficult problem. This situation is mainly due to the non-stationary of noise and poor Signal-to-Noise Ratio (SNR).

In order to alleviate this problem, we propose a *trajectory auto-segmentation and clustering* scheme for speech/non-speech labeling. Under the assumption that noise is more homogeneous but less strong than speech, sub-band energy trajectory can be divided into successive and homogeneous segments, from which the desired speech/non-speech labeling can be obtained by clustering the energy means of those segments. Using the labeled non-speech samples, noise spectrum can be estimated with both time and frequency resolution, and spectral subtraction can thus be performed to get a rough estimate of the clean speech trajectories.

After spectral substraction, the noise subtracted trajectories only approximate the static "level" of the clean speech, without considering its dynamic "shape". Conventional spectral subtraction methods finish here, and extract feature coefficients directly from these rough trajectories. As a result, the information we observed from the noisy speech signal has not been fully exploited. In fact, there is an explicit constraint between static and dynamic features of a trajectory. As previous research [4] has shown that the dynamic feature is more robust than static feature in noisy environment, it is then feasible to refine the noise subtracted trajectory, by using the observed dynamic information of the original speech. For doing this, we propose a *trajectory regeneration* scheme using trajectory HMMs [5], to impose the relationship between static and dynamic features. The static-dynamic constraint results in a better estimate of the clean speech trajectory in our study, when compared with using spectral subtraction method only.

The rest of this paper is organized as follows. Section 2 introduces the trajectory auto-segmentation and clustering scheme for noise estimation and spectral subtraction. Section 3 discusses the details of trajectory regeneration using static-dynamic constraint. Algorithm implementation and experimental results are reported in Section 4. And finally we draw our conclusions in Section 5.

2 Spectral Subtraction Using Trajectory Auto-segmentation and Clustering

2.1 Trajectory Auto-segmentation and Clustering

We propose an auto-segmentation and clustering scheme to label speech and non-speech regions along a trajectory, which is then used as the guidance for noise estimate and spectral subtraction. This method is effectual mainly because of the helpful nature of the noise corrupted trajectory. The solid line in Fig. 1(A) illustrates a sub-band energy trajectory in 10dB SNR car noise (the dash line for clean speech trajectory is also drawn for reference). We can see from the

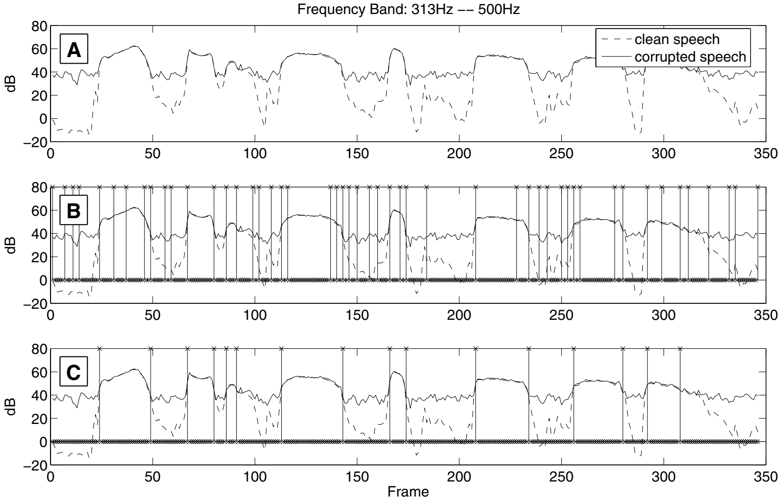


Fig. 1. Typical behavior of trajectory auto-segmentation and clustering in noisy environment (sub-band energy trajectory of "O73643O" corrupted by car noise from the Aurora 2 database, 10dB SNR)

figure that: 1) neighboring non-speech regions tend to be more homogeneous, or less variable than speech along time; 2) the transition between speech and non-speech usually leads to a sudden change on trajectory, and 3) noise mainly corrupts background silence and unvoiced regions of speech, leaving most of the higher volume, voiced regions less corrupted.

Based on these properties of the corrupted speech trajectory, we can first divide a trajectory into certain number of homogeneous segments, and then cluster them into two classes according to their segment energies. Because we assume that the energy of a voiced speech segment is always higher than that of the non-speech's, it is quite straightforward to label the segments in the higher energy cluster as speech, and the others as non-speech. Formally, the trajectory auto-segmentation and clustering scheme can be described as follows:

Given an energy trajectory of T frames, first, we divided it into M successive segments, say, $\Phi = \phi_1, \phi_2, \ldots, \phi_M$, to make each segment as homogeneous as possible. M is chosen to be proportional to the length of the utterance according to a fixed ratio on the basis of experience (typically $M = \lfloor T/5 \rfloor$ or $M = \lfloor T/10 \rfloor$). We measure the homogeneity of a segment ϕ_m by the energy variance $\mathrm{var}[E(\phi_m)]$ within the segment. Consequently, the optimal trajectory segmentation Φ_{opt} can be obtained by the following optimization that:

$$\Phi_{\mathrm{opt}} = \mathrm{argmin}_{\Phi} \sum_{m=1}^{M} \mathrm{var}[E(\phi_m)] \, / \, M \tag{1}$$

It is efficient to optimize Eq. (1) using dynamic programming. The typical behavior of auto-segmentation on a corrupted trajectory is shown in Fig. 1(B).

As we can see from the figure, once there is a sudden change on the trajectory, the algorithm will put a segment boundary to maintain homogeneity within the segment. This property provides a way to separate speech from non-speech, because neighboring high-energy segments can be identified and clustered as speech. A simple k-means clustering with respect to the segment energies is applied after auto-segmentation. These neighboring segments in the cluster with a higher energy level are merged and labeled as speech, while the others are labeled as non-speech. The final clustering result is shown in Fig. 1(C), in which the clustered regions 2, 4, 6, 8, etc, have been labeled as speech.

Sub-band trajectory auto-segmentation and clustering labels speech/non-speech regions along the time axis. Therefore, this method can be performed on an appropriate number of trajectories from different sub-bands, in order to obtain the speech/non-speech labeling with both time and frequency resolutions. This is the basis for noise estimate and spectral subtraction in latter process.

2.2 Trajectory Spectral Subtraction

Spectral subtraction is carried out for noise suppression, on trajectories of each frequency bin. Given a trajectory of frequency ω and its corresponding sub-band speech/non-speech labeling, we use the samples which have been labeled as non-speech to get a smooth estimate of the noise $\widetilde{N}(\omega, t)$:

$$\widetilde{N}(\omega, t) = \begin{cases} \sum\limits_{\tau=t-l}^{t+l} X(\omega, \tau)\delta(\omega, \tau) \;/\; \sum\limits_{\tau=t-l}^{t+l} \delta(\omega, \tau), & \text{if } \sum\limits_{\tau=t-l}^{t+l} \delta(\omega, \tau) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where

$$\delta(\omega, \tau) = \begin{cases} 0 & \text{for speech} \\ 1 & \text{for non-speech} \end{cases} \tag{3}$$

and $X(\omega, t)$, $2l{+}1$ are the noisy speech trajectory and the length of the smoothing window, respectively.

By subtracting noise from noisy speech, the estimate of the clean speech trajectory $\widetilde{S}(\omega, t)$ with an over-estimation factor $\alpha(\omega)$ can be formulated as:

$$\widetilde{S}(\omega, t) = \max\{X(\omega, t) - \alpha(\omega)\widetilde{N}(\omega, t),\ \beta X(\omega, t)\} \tag{4}$$

in which β is the noise-masking floor factor.

In Eq. (4), $\alpha(\omega)$ is a sub-band SNR-dependent over-estimation factor. The basic idea of this factor is to subtract more noise in low SNR regions [1]. In fact, when we use sub-band trajectory auto-segmentation and clustering, the divergence between speech and non-speech clusters is directly related to the SNR. Therefore, this divergence can be used instead of the SNR to determine the over-estimation factor for spectral subtraction.

Let μ_{s}, μ_{n}, σ_{s}^2 and σ_{n}^2 be the energy means and variances of the trajectory's speech and non-speech clusters, respectively, and assume the energy distributions within each cluster are single Gaussians, we calculate the Kullback-Leibler (K-L) divergence between speech and non-speech clusters for determining $\alpha(\omega)$:

$$D = D(\mathrm{s} \parallel \mathrm{n}) = \int_x \mathcal{N}(x \mid \mu_\mathrm{s}, \sigma_\mathrm{s}^2) \log \frac{\mathcal{N}(x \mid \mu_\mathrm{s}, \sigma_\mathrm{s}^2)}{\mathcal{N}(x \mid \mu_\mathrm{n}, \sigma_\mathrm{n}^2)} \,\mathrm{d}x \tag{5}$$

For each utterance with N sub-band trajectories of different frequencies, the speech/non-speech divergence D_n, $n = 1, \ldots, N$, can be computed. Let $D_{\min}$ and $D_{\max}$ be the minimal and maximal of D_n, and suppose ω belongs to the sub-band of the $i-$th trajectory, the over-estimation factor $\alpha(\omega)$ is defined as:

$$\alpha(\omega) = 1.5 - 0.5 \times (D_i - D_{\min})/(D_{\max} - D_{\min}) \tag{6}$$

As we can see from Eq. (6), the frequency bins in the sub-bands with low speech/non-speech divergences (which means poor SNR) get a higher over-estimation factor than the ones in the sub-bands with high divergences (which means high SNR). Each $\alpha(\omega)$ is then used in Eq. (4) to complete the spectral subtraction process.

3 Trajectory Regeneration Under Static-Dynamic Constraint

After spectral subtraction, a rough estimate of the trajectories of each frequency bin is obtained. These trajectories approximate the static “level” of the power spectrum of the clean speech, and conventional spectral subtraction methods use them directly to calculate feature coefficients for speech recognizer. However, as we essentially subtract a smooth estimate of the static noise power from the corrupted speech trajectories, the dynamic “shape” of these trajectories is in fact neglected. In fact, there is a explicit constraint between static and dynamic features. This relationship can be imposed to refine the roughly estimated trajectories so that the modified trajectories would approximate the clean speech trajectories both in “level” and “shape”.

For doing this, we propose a trajectory regeneration scheme using the idea of trajectory HMM in speech synthesis [5]. A trajectory is first divided into successive “states” using auto-segmentation, and then regenerated by solving the weighted normal equation which impose the static-dynamic constraint. The following two subsections will describe the details of the two steps.

3.1 Trajectory State Sequence by Using Auto-segmentation

The use of trajectory HMM relies on a sequence of HMM states, and each state represents both the static and dynamic features of certain number of frames. The first step of trajectory regeneration is to divide the trajectory into successive segments, and regard each segment as a trajectory state. Auto-segmentation is used again to find the optimal state boundary $\Phi_{\mathrm{opt}} = \phi_1, \phi_2, \ldots, \phi_M$ of the trajectory after spectral subtraction. Then, means and variances for both static and dynamic coefficients of all states are computed. Because previous research

[4] has shown that the dynamic feature is more robust than static feature in noisy environment, we use the noise subtracted trajectory to compute the static features of the states, while the observed noisy speech trajectory is used to compute the dynamic features (state boundary ϕ_m is kept the same for both the two cases). It can be formulated as:

$$\begin{aligned} \mu_{\phi_m}^{(\mathrm{s})} &= \underset{t \in \phi_m}{\mathrm{mean}}[\widetilde{S}(\omega, t)] & \sigma_{\phi_m}^{2(\mathrm{s})} &= \underset{t \in \phi_m}{\mathrm{var}}\,[\widetilde{S}(\omega, t)] \\ \mu_{\phi_m}^{(\mathrm{d})} &= \underset{t \in \phi_m}{\mathrm{mean}}[\Delta X(\omega, t)] & \sigma_{\phi_m}^{2(\mathrm{d})} &= \underset{t \in \phi_m}{\mathrm{var}}\,[\Delta X(\omega, t)] \end{aligned} \tag{7}$$

in which $\Delta X(\omega, t)$ is the first-order dynamic trajectory of the noisy speech, and (s), (d) stand for the static and dynamic features, respectively. Note that in Eq. (7), we only use the first-order delta trajectory as the dynamic feature for simpleness. The use of delta-delta feature can be implemented similarly.

By using auto-segmentation, a trajectory of T frames can be aligned by its state sequence $\boldsymbol{q} = \{q_1, \ldots, q_T\}$. For each state $\{q_t \mid t \in \phi_m\}$, the static and dynamic feature can be written as:

$$\begin{aligned} \mu_{q_t}^{(\mathrm{s})} &= \mu_{\phi_m}^{(\mathrm{s})} & \sigma_{q_t}^{2(\mathrm{s})} &= \sigma_{\phi_m}^{2(\mathrm{s})} \\ \mu_{q_t}^{(\mathrm{d})} &= \mu_{\phi_m}^{(\mathrm{d})} & \sigma_{q_t}^{2(\mathrm{d})} &= \sigma_{\phi_m}^{2(\mathrm{d})} \end{aligned} \tag{8}$$

Then the state sequence $\boldsymbol{q}$ and its static and dynamic features are used in the next step for trajectory regeneration.

3.2 Trajectory Regeneration Using Trajectory HMM

Given the state sequence $\boldsymbol{q} = \{q_1, \ldots, q_T\}$ of one trajectory, we regenerate the state output parameter sequence $\boldsymbol{O} = [\boldsymbol{o}_1^\top, \ldots, \boldsymbol{o}_T^\top]^\top$ in such a way that $P(\boldsymbol{O} \mid \boldsymbol{q})$ is maximized with respect to $\boldsymbol{O}$. When only static feature is considered, maximizing $P(\boldsymbol{O} \mid \boldsymbol{q})$ is equivalent to generate a trajectory consisting of a piecewise constant sequence of the static means of $\boldsymbol{q}$, so the dynamic information is totally discarded. This situation is due to the independence assumption of state output probabilities, therefore, the static-dynamic constraint of the state parameters need to be imposed to avoid this problem.

Let $\boldsymbol{o}_t$ be a vector consisted of the trajectory's static feature $o_t^{(\mathrm{s})}$ and its first-order delta (dynamic) feature $o_t^{(\mathrm{d})}$, that is:

$$\mathbf{o}_t = [o_t^{(\mathrm{s})}, o_t^{(\mathrm{d})}]^\top \tag{9}$$

In Eq. (9), $o_t^{(\mathrm{d})}$ can be rewritten as the weighted sum of $o_t^{(\mathrm{s})}$ in a delta window of $2L + 1$:

$$o_t^{(\mathrm{d})} = \sum_{\tau=-L}^{L} \omega(\tau) o_{t+\tau}^{(\mathrm{s})} \tag{10}$$

where $\omega(\tau)$ is the weight coefficients given by:

$$\omega(\tau) = \frac{\tau}{2 \sum_{\lambda=1}^{L} \lambda^2} \tag{11}$$

Therefore, by substituting Eq. (10) into Eq. (9), $\boldsymbol{O} = [\boldsymbol{o}_1^\top, \ldots, \boldsymbol{o}_T^\top]^\top$ can be rearrange in a matrix form so that:

$$\boldsymbol{O} = \mathbf{W}\boldsymbol{c} \tag{12}$$

in which

$$\boldsymbol{c} = [o_1^{(\mathrm{s})}, o_2^{(\mathrm{s})}, \ldots, o_T^{(\mathrm{s})}]^\top \tag{13}$$

$$\mathbf{W} = [\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_T]^\top \tag{14}$$

$$\mathbf{W}_t = [\boldsymbol{\omega_t}^{(\mathrm{s})}, \boldsymbol{\omega_t}^{(\mathrm{d})}] \tag{15}$$

and

$$\begin{aligned} \boldsymbol{\omega_t}^{(\mathrm{s})} &= [\,\underbrace{0, \ldots, 0}_{t-1}, 1, \underbrace{0, \ldots, 0}_{T-t}\,]^\top \\ \boldsymbol{\omega_t}^{(\mathrm{d})} &= [\,\underbrace{0, \ldots, 0}_{t-L-1}, \omega(-L), \ldots, \omega(+L), \underbrace{0, \ldots, 0}_{T-t-L}\,]^\top \end{aligned} \tag{16}$$

Assuming that each state output probability can be characterized by a two-dimensional single Gaussian distribution with diagonal covariance matrix, then, $P(\boldsymbol{O} \mid \boldsymbol{q})$ can be written as:

$$P(\boldsymbol{O} \mid \boldsymbol{q}) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{o}_t \mid \boldsymbol{\mu}_{q_t}, \boldsymbol{\Sigma}_{q_t}) = \mathcal{N}(\boldsymbol{O} \mid \boldsymbol{\mu}_{\boldsymbol{q}}, \boldsymbol{\Sigma}_{\boldsymbol{q}}) \tag{17}$$

in which

$$\begin{aligned} \boldsymbol{\mu}_{q_t} &= [\mu_{q_t}^{(\mathrm{s})}, \mu_{q_t}^{(\mathrm{d})}]^\top \\ \boldsymbol{\Sigma}_{q_t} &= \mathrm{diag}[\sigma_{q_t}^{2(\mathrm{s})}, \sigma_{q_t}^{2(\mathrm{d})}] \end{aligned} \tag{18}$$

and

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{q}} &= [\boldsymbol{\mu}_{q_1}^\top, \boldsymbol{\mu}_{q_2}^\top, \ldots, \boldsymbol{\mu}_{q_T}^\top]^\top \\ \boldsymbol{\Sigma}_{\boldsymbol{q}} &= \mathrm{diag}[\boldsymbol{\Sigma}_{q_1}, \boldsymbol{\Sigma}_{q_2}, \ldots, \boldsymbol{\Sigma}_{q_T}] \end{aligned} \tag{19}$$

Under the static-dynamic constraint given by Eq. (12), maximizing Eq. (17) with respect to $\boldsymbol{O}$ is equivalent to that with respect to $\boldsymbol{c}$. So by setting:

$$\frac{\partial \log P(\mathbf{W}\boldsymbol{c} \mid \boldsymbol{q})}{\partial \boldsymbol{c}} = 0 \tag{20}$$

we obtain:

$$\mathbf{W}^\top \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \mathbf{W}\boldsymbol{c} = \mathbf{W}^\top \boldsymbol{\Sigma}_{\boldsymbol{q}}^{-1} \boldsymbol{\mu}_{\boldsymbol{q}} \tag{21}$$

As a result, by solving the weighted normal equation in Eq. (21), a new trajectory in terms of its static feature sequence $\boldsymbol{c} = [o_1^{(\mathrm{s})}, o_2^{(\mathrm{s})}, \ldots, o_T^{(\mathrm{s})}]^\top$ can be regenerated. The refined trajectory follows not only the clean speech "level" (which is estimated by spectral subtraction), but also its dynamic "shape" (which is observed from the noisy speech trajectory). This balance between static and dynamic features in the maximum likelihood sense thus yields a better approximation of the clean speech trajectory.

4 Experiments

4.1 Implementations

The corrupted speech data is processed in a time window of 25ms, shifted every 10ms. A pre-emphasis with a coefficient of 0.97 is performed. The speech samples are then hamming windowed, and transformed with a 256-point FFT. Then, the power spectrum of each frame is calculated for noise suppression.

The frequency range between 64Hz and Nyquist frequency (4kHz) is equally divided into 23 half-overlapped sub-bands in Mel-frequency. Sub-band energy is then calculated and 23 sub-band energy trajectories are formed. Auto-segmentation and clustering are used to divide each trajectory into segments and classify them speech or non-speech. Meanwhile, the K-L divergence between speech and non-speech clusters is calculated. This divergence is then used to determine the factor $\alpha(\omega)$ in Eq. (6).

Spectral subtraction is applied to the trajectories of every frequency bin. Because each frequency bin usually belongs to two of the 23 sub-bands (as every two neighboring sub-bands are half-overlapped), an average between the two corresponding sub-bands is needed when their speech/non-speech labeling are different. Following this way, noise power is estimated by calculating a weighted average, using the non-speech samples in a 500ms window. It is then subtracted from the corrupted trajectory using Eq. (4).

After spectral subtraction, we get one roughly modified trajectory for each frequency bin. Auto-segmentation is used again for the second time, to divide each trajectory into successive segments. Means and variances of both static and dynamic features of the segments are computed, and trajectory states are formed. After that, Eq. (21) is solved for trajectory regeneration, and the constraint between static and dynamic features is naturally imposed. Due to the special structure of $\mathbf{W}^{\top}\mathbf{\Sigma}_{\boldsymbol{q}}^{-1}\mathbf{W}$, Eq. (21) can be solved effectively by Cholesky decomposition.

In the last step, acoustic coefficients for the recognizer are calculated from the regenerated trajectories. The final output of the algorithm is 12 mel-scaled cepstral coefficients plus log energy, along with their delta and acceleration coefficients. As a result, a 39-dimensional feature vector, i.e., MFCC_E_D_A, is used as the input to the speech recognizer.

4.2 Experimental Setup

The recognition experiments were carried out on the Aurora 2 database using HTK [6]. Both training and testing were performed with ETSI provided scripts [7]. Fig. 2 illustrates the models and noise compensation methods we built and compared in our experiments: (A) the baseline feature extraction method without noise compensation (ETSI FE v2.0); (B) spectral subtraction without trajectory regeneration, and (C) spectral subtraction followed by trajectory regeneration. Because we mainly focus on evaluating our algorithm for the performance of estimating clean speech trajectory, no feature normalization methods nor any other special treatment (e.g., VAD) was performed.

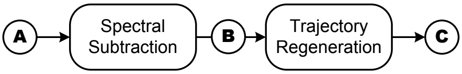


Fig. 2. Comparation of noise compensation methods: (A) baseline without compensation, (B) spectral substration, and (C) trajectory regeneration

4.3 Experimental Results

The experimental results are given in Fig. 3 and Table. 1. For all the three testing sets and all the SNRs between 20dB to 0dB, a consistent recognition error reduction can be observed for both methods. These results show that the noise compensation improves recognition performance with or without trajectory regeneration.

For spectral subtraction based methods, the best recognition improvement is often achieved when the mismatch between training and testing conditions is very high. In our experiments, both methods achieve a better performance improvement in clean training than multi-condition training. Spectral subtraction without trajectory regeneration relatively improves the performance by 23.21% for clean training, and only 5.58% for multi-condition training.

When comparing with using spectral subtraction only, spectral subtraction along with trajectory regeneration exhibits the best recognition performance. Word error rate is relatively reduced by 42.59% for clean training, and 15.80% for multi-condition training after trajectory regeneration. This result shows that the trajectory refinement by imposing the static-dynamic constraint leads to a better estimate of the clean speech trajectory. The average absolute performance achieved by the proposed trajectory regeneration method for clean and multi-condition training is 82.81%, which relatively improves the baseline system by 29.19%.

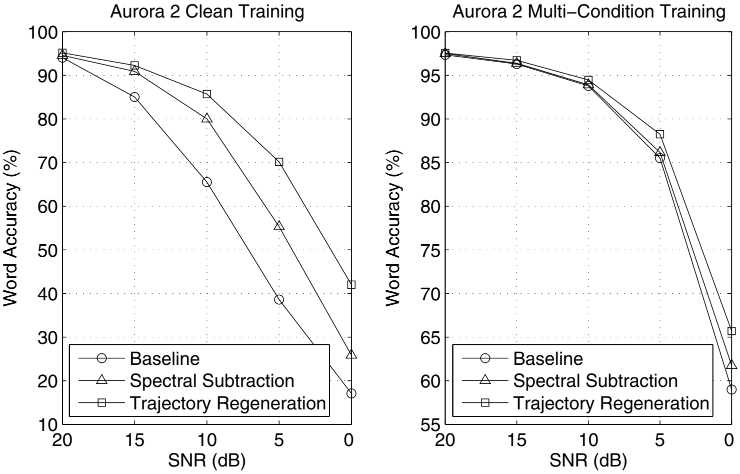


Fig. 3. Average word accuracy against SNR for the baseline, spectral subtraction and trajectory regeneration

Table 1. Recognition performance summaries

Absolute performance					
Methods	Training Mode	Set A	Set B	Set C	Overall
Baseline	Clean Only	61.34	55.74	66.14	60.06
	Multi-condition	87.81	86.27	83.77	86.39
	Average	74.58	71.01	74.96	**73.23**
Spectral Subtraction	Clean Only	70.01	66.93	72.76	69.33
	Multi-condition	88.63	87.12	84.21	87.15
	Average	79.32	77.03	78.49	**78.24**
Trajectory Regeneration	Clean Only	78.58	75.29	77.64	77.07
	Multi-condition	89.72	88.26	86.75	88.54
	Average	84.15	81.78	82.20	**82.81**
Performance relative to the baseline					
Methods	Training Mode	Set A	Set B	Set C	Overall
Spectral Subtraction	Clean Only	22.43%	25.28%	19.55%	23.21%
	Multi-condition	6.73%	6.19%	2.71%	5.58%
	Average	14.58%	15.74%	11.13%	**14.40%**
Trajectory Regeneration	Clean Only	44.59%	44.17%	33.96%	42.59%
	Multi-condition	15.67%	14.49%	18.36%	15.80%
	Average	30.13%	29.33%	26.16%	**29.19%**

5 Conclusions

In this paper we evaluate our noise compensation front-end algorithm based on signal trajectory processing. The proposed algorithm consists of two stages, which at first subtract estimated noise power from the corrupted speech trajectory, and then regenerate the trajectory using an explicit static-dynamic constraint. Our experiments indicate that a reliable noise estimate can be obtained by using trajectory auto-segmentation and clustering. And it is feasible to refine the noise subtracted trajectory, by considering its dynamic feature. Spectral subtraction along with trajectory regeneration achieves the best recognition performance in our experiments on the Aurora 2 database. This result also reconfirms that the dynamic feature is more robust than static feature in noisy environment. How to exploit the constraint and balance between static and dynamic features of a trajectory for improved recognition performance will be our future work.

References

1. P. Lockwood and J. Boudy, “Experiments with a Nonlinear Spectral Subtractor (NSS), Hidden Markov Models and the projection, for robust speech recognition in cars,” *Speech Communication*, Vol. 11, pp. 215-228, 1992.

2. D. Ealey, H. Kelleher, and D. Pearce, “Harmonic Tunneling: Tracking Non-Stationary Noises During Speech,” in *Proc. Eurospeech 2001*, Scandinavia, pp. 437-440, 2001.
3. N. W. D. Evans and J. S. Mason, “Time-Frequency Quantile-Based Noise Estimation,” in *Proc. EUSIPCO*, Toulouse, Vol.1, pp. 539-542, 2002.
4. C. Yang, F. K. Soong, and T. Lee, “Static and Dynamic Spectral Features: Their Noise Robustness and Optimal Weights for ASR,” in *Proc. ICASSP 2005*, Philadelphia, Vol. 1, pp. 241-244, 2005.
5. K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, “Speech parameter generation algorithms for HMM-based speech synthesis,” in *Proc. ICASSP 2000*, Istanbul, Vol.3, pp. 1315-1318, 2000.
6. S. J. Young, G. Evermann, et al., *The HTK Book*, Revised for HTK Version 3.3, 2005.
7. H. G. Hirsch and D. Pearce, “The Aurora Experimental Framework for the Performance Evaluation of Speech Recognition under Noisy Conditions,” *ISCA ITRW ASR 2000*, Paris, pp.181-188, 2000.

An HMM Compensation Approach Using Unscented Transformation for Noisy Speech Recognition*

Yu Hu[1,2] and Qiang Huo[1]

[1] Department of Computer Science,
The University of Hong Kong, Hong Kong
[2] Department of Electronic Engineering & Information Science,
University of Science and Technology of China, Hefei
jadefox@ustc.edu, qhuo@cs.hku.hk

Abstract. The performance of current HMM-based automatic speech recognition (ASR) systems degrade significantly in real-world applications where there exist mismatches between training and testing conditions caused by factors such as mismatched signal capturing and transmission channels and additive environmental noises. Among many approaches proposed previously to cope with the above robust ASR problem, two notable HMM compensation approaches are the so-called Parallel Model Combination (PMC) and Vector Taylor Series (VTS) approaches, respectively. In this paper, we introduce a new HMM compensation approach using a technique called Unscented Transformation (UT). As a first step, we have studied three implementations of the UT approach with different computational complexities for noisy speech recognition, and evaluated their performance on Aurora2 connected digits database. The UT approaches achieve significant improvements in recognition accuracy compared to log-normal-approximation-based PMC and first-order-approximation-based VTS approaches.

1 Introduction

Most of current ASR systems use MFCCs (Mel-Frequency Cepstral Coefficients) and their derivatives as speech features, and a set of Gaussian mixture continuous density HMMs (CDHMMs) for modeling basic speech units. It is well known that the performance of such an ASR system trained with clean speech will degrade significantly when the testing speech is distorted by additive noises. How to achieve the noise robustness in the above scenario has been an important research topic in ASR field. Among many approaches proposed previously to cope with the above robust ASR problem, two notable HMM compensation approaches are the so-called Parallel Model Combination (PMC) (e.g., [2]) and

* This work was done while Y. Hu worked at The University of Hong Kong, and was supported by grants from the RGC of the Hong Kong SAR (Project No. HKU 7039/02E) and Anhui USTC iFLYTEK Co. Ltd., Hefei, China.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 346–357, 2006.

Vector Taylor Series (VTS) (e.g., [7,8,6,1]) approaches, respectively. For both approaches, the following simplified assumptions are made: 1) The speech and noise signals are independent, and additive in the time domain; 2) The alignment between a speech frame and the corresponding Gaussian component of CDHMM used to train the speech models from the clean speech data is not altered by the addition of noise; 3) The Gaussian PDF (probability density function) remains appropriate for modeling the feature vectors of noisy speech aligned to the corresponding component. Having made the above assumptions, the problem of HMM compensation is simplified as a problem of how to calculate the mean vector and covariance matrix, $\{\mu_{\mathbf{y}}, \Sigma_{\mathbf{y}}\}$, for each Gaussian component of the noisy speech from the corresponding statistics $\{\mu_{\mathbf{x}}, \Sigma_{\mathbf{x}}\}$ for clean speech and $\{\mu_{\mathbf{n}}, \Sigma_{\mathbf{n}}\}$ for noise, respectively.

Even after the above simplification, the expressions for estimating the corrupted speech model parameters do not have closed-form solutions. Various approximations have previously been proposed to solve this problem. For PMC approach, *numerical integration* and *data-driven PMC* are two techniques that provide an accurate approximation but is computationally expensive, while the so-called *log-normal* and *log-add* approximations are less accurate but computationally more efficient [2]. For VTS approach [7,8], a truncated (typically up to the first-order) Taylor series expansion is used to approximate the nonlinear distortion function that relates feature vectors of noisy speech to the ones of clean speech and noise. In this paper, we propose a new approximation approach to address the above problem by using a technique called Unscented Transformation (UT) [5] developed originally in the field of automatic control for improving the Extended Kalman Filtering (EKF) technique based on the first-order VTS approximation of the relevant nonlinear functions. We will demonstrate that this new approach offers a better performance than that of both PMC and VTS approaches.

The rest of the paper is organized as follows. In Section 2, we introduce the basic concept of UT and show how we can use this tool to formulate three different solutions to HMM compensation for noisy speech recognition. Then, two important implementation issues are discussed in Section 3. We will compare the accuracy of different approximation methods via simulation experiments in Section 4. The experimental results on Aurora2 database are reported and analyzed in Section 5. Finally, we conclude the paper in Section 6.

2 HMM Compensation Using Unscented Transformation

2.1 Some Notations and the Basic Formulation of UT

In this study, only additive noise is considered and the effect of channel distortion is ignored. Therefore, the distortion function between clean and noisy speech in log-spectral domain can be expressed as:

$$\mathbf{Y}^{log} = \mathbf{f}(\mathbf{X}^{log}, \mathbf{N}^{log}) = log\Big(exp(\mathbf{X}^{log}) + exp(\mathbf{N}^{log})\Big) , \tag{1}$$

where $\mathbf{Y}$, $\mathbf{X}$ and $\mathbf{N}$ are random feature vectors for noisy speech, clean speech and additive noise, respectively; and the superscript *log* indicates that the above feature vectors are extracted in the log-spectral domain. During HMM compensation, we need to estimate the mean vector $\mu_{\mathbf{y}}^{cep}$ and covariance matrix $\Sigma_{\mathbf{y}}^{cep}$ for each Gaussian component defined as

$$\mu_{\mathbf{y}}^{cep} = \mathcal{E}\left\{\mathbf{Y}^{cep}\right\}, \ \ \Sigma_{\mathbf{y}}^{cep} = \mathcal{E}\left\{(\mathbf{Y}^{cep} - \mu_{\mathbf{y}}^{cep})(\mathbf{Y}^{cep} - \mu_{\mathbf{y}}^{cep})^T\right\}, \tag{2}$$

where the expectations are taken over the PDF of the noisy speech random feature vector $\mathbf{Y}^{cep}$, the superscript *cep* indicates that the relevant items are in cepstral domain, and $(\cdot)^T$ denotes the transpose of a vector or a matrix hereinafter. For convenience, estimation will be performed first in the log-spectral domain and then be transformed back to the cepstral domain. We then have

$$\mu_{\mathbf{y}}^{cep} = LC\mu_{\mathbf{y}}^{log}; \quad \Sigma_{\mathbf{y}}^{cep} = LC\Sigma_{\mathbf{y}}^{log}C^TL^T \tag{3}$$

with

$$\mu_{\mathbf{y}}^{log} = \mathcal{E}\Big\{log(exp(\mathbf{X}^{log}) + exp(\mathbf{N}^{log}))\Big\} \tag{4}$$

$$\begin{aligned}\Sigma_{\mathbf{y}}^{log} = \mathcal{E}\Big\{&\Big(log(exp(\mathbf{X}^{log}) + exp(\mathbf{N}^{log})) - \mu_{\mathbf{y}}^{log}\Big) \cdot \\ &\Big(log(exp(\mathbf{X}^{log}) + exp(\mathbf{N}^{log})) - \mu_{\mathbf{y}}^{log}\Big)^T\Big\},\end{aligned} \tag{5}$$

where C is the matrix for DCT transformation, and L is the matrix for cepstral lifting operation, whose exact definition can be found in, e.g. [9]. Given the PDF parameters, $\mu_{\mathbf{x}}^{cep}$ and $\Sigma_{\mathbf{x}}^{cep}$, for clean speech in cepstral domain, the corresponding PDF parameters in log-spectral domain can be obtained as follows:

$$\mu_{\mathbf{x}}^{log} = C^{-1}L^{-1}\mu_{\mathbf{x}}^{cep}, \quad \Sigma_{\mathbf{x}}^{log} = C^{-1}L^{-1}\Sigma_{\mathbf{x}}^{cep}(L^{-1})^T(C^{-1})^T . \tag{6}$$

Similar operations can be used to derive PDF parameters $\mu_{\mathbf{n}}^{log}$ and $\Sigma_{\mathbf{n}}^{log}$ for noise in the log-spectral domain from the PDF in the cepstral domain that is assumed to follow a normal distribution with a mean vector $\mu_{\mathbf{n}}^{cep}$ and covariance matrix $\Sigma_{\mathbf{n}}^{cep}$. In the following, we will formulate our problem in the log-spectral domain, therefore the superscript *log* will be omitted if no confusion will be caused according to the context of relevant discussions.

Now, let's explain what is the unscented transformation (UT) [5]. Suppose a random vector $\mathbf{X}$ has a known mean $\mu_{\mathbf{x}}$ and covariance $\Sigma_{\mathbf{x}}$. A second random vector $\mathbf{Y}$ is related to $\mathbf{X}$ via a nonlinear function $\mathbf{Y} = \mathbf{f}(\mathbf{X})$. The UT was developed as an effective method to calculate the mean $\mu_{\mathbf{y}}$ and covariance $\Sigma_{\mathbf{y}}$ for $\mathbf{Y}$, and works as follows: First, a set of points (the so-called *sigma points*) are chosen such that their sample mean and covariance equal to the $\mu_{\mathbf{x}}$ and $\Sigma_{\mathbf{x}}$, respectively. A weight is also specified for each sigma point. Then, the nonlinear function $\mathbf{f}(\cdot)$ is applied to each sigma point, in turn, to yield a set of transformed points. Finally, the mean and covariance of $\mathbf{Y}$ is estimated as the weighted average and the weighted outer product of the transformed points, respectively. Readers are referred to [5] for more details of UT formulation.

In the following three subsections, we show how we can use the UT technique to formulate three different solutions to HMM compensation for noisy speech recognition.

2.2 Method 1: UT with Augmented Random Vectors

Given $\mathbf{X}$ and $\mathbf{N}$, we can form an augmented K_a-dimensional random vector, $\mathbf{X}^a = [\mathbf{X}^T, \mathbf{N}^T]^T$, where $K_a = K_\mathbf{x} + K_\mathbf{n}$ with $K_\mathbf{x}$ and $K_\mathbf{n}$ being the dimensions of $\mathbf{X}$ and $\mathbf{N}$ respectively. In our case here, $K_\mathbf{x} = K_\mathbf{n} = K$. Consequently, the nonlinear distortion function in Eq. (1) can be rewritten as a function of $\mathbf{X}^a$:

$$\mathbf{Y} = \mathbf{f}^a(\mathbf{X}^a) = \mathbf{f}(\mathbf{X}, \mathbf{N}). \tag{7}$$

Let's use μ^a and Σ^a to denote the mean vector and covariance matrix of $\mathbf{X}^a$ respectively. Then we have

$$\mu^a = [{\mu_\mathbf{x}}^T, {\mu_\mathbf{n}}^T]^T, \qquad \Sigma^a = \begin{bmatrix} \Sigma_\mathbf{x} & \mathbf{0}_{K\times K} \\ \mathbf{0}_{K\times K} & \Sigma_\mathbf{n} \end{bmatrix} . \tag{8}$$

Given the above new notations, the following standard UT procedure can be used to calculate $\mu_\mathbf{y}$ and $\Sigma_\mathbf{y}$ for $\mathbf{Y}$:

1. Create a set of sigma points S, $S = \{\chi_j^a, W_j^a; j = 0, 1, ..., 4K\}$, which consists of $4K + 1$ K_a-dimensional vectors χ_j^a's and their associated weights W_j^a's, as follows:

$$\begin{aligned} \chi_0^a &= \mu^a, \quad W_0^a = \frac{\kappa^a}{(2K + \kappa^a)}; \\ \chi_i^a &= \mu^a + \left(\sqrt{(2K + \kappa^a)\Sigma^a}\right)_i, \quad W_i^a = \frac{1}{2(2K + \kappa^a)}; \\ \chi_{i+2K}^a &= \mu^a - \left(\sqrt{(2K + \kappa^a)\Sigma^a}\right)_i, \quad W_{i+2K}^a = \frac{1}{2(2K + \kappa^a)} ; \end{aligned} \tag{9}$$

   where $i = 1, \ldots, 2K$; κ^a is a control parameter whose effect is explained in [5]; $(\sqrt{\Sigma})_i$ is the ith column of the square root matrix Θ of Σ, if $\Sigma = \Theta\Theta^T$.
2. Transform each of the above sigma points by using the nonlinear function $\mathbf{f}^a(\cdot)$ to generate the set of transformed sigma points:

$$\gamma_i = \mathbf{f}^a(\chi_i^a) . \tag{10}$$

3. The mean of random vector $\mathbf{Y}$ is given by the weighted average of the transformed sigma points

$$\mu_\mathbf{y} = \sum_{i=0}^{4K+1} W_i^a \gamma_i , \tag{11}$$

   where $\sum_{i=0}^{4K+1} W_i^a = 1$ by definition.

4. The covariance of random vector **Y** is given by the weighted outer product of the transformed sigma points

$$\Sigma_{\mathbf{y}} = \sum_{i=0}^{4K+1} W_i^a \left\{\gamma_i - \mu_{\mathbf{y}}\right\} \left\{\gamma_i - \mu_{\mathbf{y}}\right\}^T \quad . \tag{12}$$

For the augmented formulation given in Eq. (8), the formulas for generating the sigma points in Step 1 of the above procedure can be derived and simplified as follows:

$$\begin{aligned}
\chi_0^a &= \begin{bmatrix} \mu_{\mathbf{x}} \\ \mu_{\mathbf{n}} \end{bmatrix}, \\
\chi_i^a &= \begin{bmatrix} \mu_{\mathbf{x}} + \left(\sqrt{(2K+\kappa^a)\Sigma_{\mathbf{x}}}\right)_i \\ \mu_{\mathbf{n}} \end{bmatrix}, \\
\chi_{i+2K}^a &= \begin{bmatrix} \mu_{\mathbf{x}} - \left(\sqrt{(2K+\kappa^a)\Sigma_{\mathbf{x}}}\right)_i \\ \mu_{\mathbf{n}} \end{bmatrix}, \\
\chi_{i+3K}^a &= \begin{bmatrix} \mu_{\mathbf{x}} \\ \mu_{\mathbf{n}} + \left(\sqrt{(2K+\kappa^a)\Sigma_{\mathbf{n}}}\right)_i \end{bmatrix}, \\
\chi_{i+4K}^a &= \begin{bmatrix} \mu_{\mathbf{x}} \\ \mu_{\mathbf{n}} - \left(\sqrt{(2K+\kappa^a)\Sigma_{\mathbf{n}}}\right)_i \end{bmatrix};
\end{aligned} \tag{13}$$

where $i = 1, \ldots, K$; and the corresponding weights are the same as in Eq. (9).

The computational costs of the UT are proportional to the number of sigma points, which depends on the dimension of the random vector concerned. In the above augmented formulation, the dimension of the augmented vector is double of that of the original individual random vectors. Naturally one want to know whether it is possible to come out a procedure that is computationally more efficient, yet still able to take advantage of the capability offered by UT. In the following subsection, we propose such a method.

2.3 Method 2: A Hybrid Approach Using UT and VTS Approximations

Eq. (1) can be rewritten as

$$\mathbf{Y} = log(exp(\mathbf{X}) + exp(\mathbf{N})) = \mathbf{N} + log(\mathbf{1} + exp(\mathbf{X} - \mathbf{N})), \tag{14}$$

where **X** and **N** are two independent K-dimensional random vectors with normal PDFs, and **1** is a K-dimensional constant vector with all elements being 1. Let's define a new random vector $\mathbf{V} = \mathbf{X} - \mathbf{N}$. Apparently, **V** has a normal PDF with mean $\mu_{\mathbf{v}} = \mu_{\mathbf{x}} - \mu_{\mathbf{n}}$ and covariance $\Sigma_{\mathbf{v}} = \Sigma_{\mathbf{x}} + \Sigma_{\mathbf{n}}$. Let's further define another new random vector

$$\mathbf{Z} = \mathbf{f}(\mathbf{V}) = log(1 + exp(\mathbf{V})). \tag{15}$$

Then, the mean and covariance of $\mathbf{Y}$ can be calculated as

$$\mu_{\mathbf{y}} = \mu_{\mathbf{n}} + \mu_{\mathbf{z}} \ , \tag{16}$$
$$\Sigma_{\mathbf{y}} = \Sigma_{\mathbf{n}} + \Sigma_{\mathbf{z}} + Cov(\mathbf{N}, \mathbf{Z}) + Cov(\mathbf{Z}, \mathbf{N}) \ . \tag{17}$$

Again, $\mu_{\mathbf{z}}$ and $\Sigma_{\mathbf{z}}$ can be calculated by using the standard UT procedure with the set of sigma points computed from $\mu_{\mathbf{v}}$ and $\Sigma_{\mathbf{v}}$ as follows:

$$\begin{aligned}
\chi_0 &= \mu_{\mathbf{x}} - \mu_{\mathbf{n}}, \quad W_0 = \frac{\kappa}{(K+\kappa)} \ ; \\
\chi_i &= \mu_{\mathbf{x}} - \mu_{\mathbf{n}} + \left(\sqrt{(K+\kappa)(\Sigma_{\mathbf{x}} + \Sigma_{\mathbf{n}})}\right)_i, \quad W_i = \frac{1}{2(K+\kappa)}; \\
\chi_{i+K} &= \mu_{\mathbf{x}} - \mu_{\mathbf{n}} - \left(\sqrt{(K+\kappa)(\Sigma_{\mathbf{x}} + \Sigma_{\mathbf{n}})}\right)_i, \quad W_{i+K} = \frac{1}{2(K+\kappa)};
\end{aligned} \tag{18}$$

where $i = 1, \ldots, K$. Note that only $2K+1$ sigma points are required here. This represents a significant reduction of computations in comparison with the UT procedure in the previous subsection that requires $4K+1$ sigma points with a double dimension of relevant vectors. Given $\mu_{\mathbf{n}}$ and $\mu_{\mathbf{z}}$, $\mu_{\mathbf{y}}$ can be calculated by using Eq. (16) accordingly.

In order to estimate $Cov(\mathbf{N}, \mathbf{Z})$ and $Cov(\mathbf{Z}, \mathbf{N})$, we approximate the nonlinear function in Eq. (15) with a truncated first-order Taylor series expansion as

$$\mathbf{Z} \approx A(\mathbf{X} - \mu_{\mathbf{x}}) - A(\mathbf{N} - \mu_{\mathbf{n}}) + log(\mathbf{1} + exp(\mu_{\mathbf{x}} - \mu_{\mathbf{n}})), \tag{19}$$

where A is a diagonal matrix whose diagonal elements are given by

$$a_{ii} = \frac{\eta_i}{1 + \eta_i} \tag{20}$$

with $\eta_i = \frac{exp(\mu_{\mathbf{x}}[i])}{exp(\mu_{\mathbf{n}}[i])}$, where $\mu_{\mathbf{x}}[i]$ is the i-th element of the mean vector of $\mathbf{X}$ corresponding to the i-th filter bank output in log-spectral domain. Then, it can be derived that

$$\begin{aligned}
Cov(\mathbf{N}, \mathbf{Z}) &\approx Cov\Big(\mathbf{N}, A(\mathbf{X} - \mu_{\mathbf{x}}) - A(\mathbf{N} - \mu_{\mathbf{n}})\Big) \\
&= Cov\Big(\mathbf{N}, A(\mathbf{X} - \mu_{\mathbf{x}})\Big) - Cov\Big(\mathbf{N}, A(\mathbf{N} - \mu_{\mathbf{n}})\Big) \\
&= -Cov(\mathbf{N}, \mathbf{N})A^T = -\Sigma_{\mathbf{n}} A^T \ ,
\end{aligned} \tag{21}$$
$$Cov(\mathbf{Z}, \mathbf{N}) \approx -A\Sigma_{\mathbf{n}} \ . \tag{22}$$

Finally, $\Sigma_{\mathbf{y}}$ can be approximated as

$$\Sigma_{\mathbf{y}} \approx \Sigma_{\mathbf{n}} + \Sigma_{\mathbf{z}} - \Sigma_{\mathbf{n}} A^T - A\Sigma_{\mathbf{n}} \ . \tag{23}$$

Therefore, the above method is referred to as a *hybrid approach* using UT and VTS approximations.

2.4 Method 3: UT Using Spherical Simplex Sigma Point Set

Actually, the above two UT-based methods can be made even more computationally efficient by using a reduced set of the so-called *spherical simplex sigma points* proposed in [4]. In this study, we only apply the above technique to improve the UT procedure in Method 2 described in the previous subsection. The sigma points are created for the random vector $\mathbf{V} = \mathbf{X} - \mathbf{N}$ as follows:

1. Specify a value for W_0 such that $0 \leq W_0 \leq 1$. Other weights are calculated as $W_i = (1 - W_0)/(K + 1)$, for $i = 1, \ldots, K + 1$.
2. Initialize vector sequence as:

$$\chi_0^1 = [0], \quad \chi_1^1 = \left[-\frac{1}{\sqrt{2W_1}}\right], \quad \chi_2^1 = \left[\frac{1}{\sqrt{2W_1}}\right] . \tag{24}$$

3. Expand vector sequence for $j = 2, \ldots, K$ according to

$$\chi_i^j = \begin{cases} \begin{bmatrix} \chi_0^{j-1} \\ 0 \end{bmatrix} & for\ i = 0 \\[2ex] \begin{bmatrix} \chi_i^{j-1} \\ -\frac{1}{\sqrt{j(j+1)W_i}} \end{bmatrix} & for\ i = 1, \ldots, j \\[2ex] \begin{bmatrix} \mathbf{0}_{j-1} \\ \frac{j}{\sqrt{j(j+1)W_i}} \end{bmatrix} & for\ i = j + 1 \end{cases} \tag{25}$$

   where $\mathbf{0}_{j-1}$ is a $(j - 1)$-dimensional vector whose elements are all zeros.
4. The set of sigma points, $\Xi = \{\xi_i; i = 0, \ldots, K + 1\}$ are finally generated as follows:

$$\xi_i = \mu_{\mathbf{v}} + \sqrt{\Sigma_{\mathbf{v}}}\chi_i^K = \mu_{\mathbf{x}} - \mu_{\mathbf{n}} + \sqrt{\Sigma_{\mathbf{x}} + \Sigma_{\mathbf{n}}}\chi_i^K . \tag{26}$$

Consequently, only $K + 2$ sigma points are required. The remaining UT steps are similar to Eqs. (10) $\sim$ (12).

3 Implementation Issues

3.1 Efficient Implementation for Calculating Square Root Matrix

Like in most of current ASR systems, we use a 39-dimensional feature vector in this study, among them, 13 features are truncated MFCCs calculated from the outputs of 23 filter banks in log-spectral domain. Therefore, an exact mapping from the cepstral domain to the log-spectral domain is not possible. By following the practice in [2], both the mean vector and diagonal covariance matrix are zero padded up to the same dimension as the number of filter banks in the log-spectral domain. When transforming the compensated mean and covariance matrix of the noisy speech in the log-spectral domain back to the cepstral domain, the relevant dimensions corresponding to higher order cepstral parameters will be truncated again and the covariance matrix in the cepstral domain is forced to be diagonal. Therefore, all the covariance matrices are assumed to be diagonal in the cepstral domain for clean speech, noise, and noisy speech. Consequently, the square root matrix for the relevant covariance matrix, Σ^{log}, in the log-spectral domain can be easily calculated from the corresponding diagonal covariance matrix, $\Sigma^{cep} = diag\{(\sigma_{ii}^{cep})^2\}$, in the cepstral domain by the following *direct method*:

$$\begin{aligned}\sqrt{\Sigma^{log}} &= \sqrt{C^{-1}L^{-1}\Sigma^{cep}(L^{-1})^T(C^{-1})^T} \\ &= C^{-1}L^{-1}\begin{bmatrix}\sigma_{11}^{cep} & 0 & & 0 \\ & \ddots & & \\ 0 & & \sigma_{K^{cep}K^{cep}}^{cep} & \\ & & & \ddots \\ 0 & & & 0\end{bmatrix}_{K^{log}\times K^{log}}\end{aligned} \tag{27}$$

where $K^{cep} = 13$ is the number of static MFCC features in the cepstral domain, and $K^{log} = 23$ is the dimension of random vectors in the log-spectral domain. The square root matrices in Eq. (13), Eq. (18), and Eq. (26) can all be calculated efficiently by the above *direct method.* Another benefit is that the number of unique sigma points will be a function of K^{cep} instead of K^{log}, leading to a great reduction of relevant computations.

3.2 Dealing with Dynamic Parameters

The proposed UT approaches can only be applied to compensating for the relevant CDHMM parameters corresponding to static MFCC features. We have not figured out a satisfactory way yet to deal with the compensation of the relevant CDHMM parameters corresponding to dynamic features. We therefore borrow a heuristic method described in [1] to compensate for only the CDHMM mean parameters corresponding to the dynamic features as follows:

$$\mu_{\Delta\mathbf{y}} \approx (I - A)\mu_{\Delta\mathbf{x}}, \quad \mu_{\Delta^2\mathbf{y}} \approx (I - A)\mu_{\Delta^2\mathbf{x}} \; ; \tag{28}$$

where A is the same as in Eq. (20), $\mu_{\Delta\mathbf{x}}$ and $\mu_{\Delta^2\mathbf{x}}$ are the mean vectors for "delta" and "delta-delta" features of clean speech, and $\mu_{\Delta\mathbf{y}}$ and $\mu_{\Delta^2\mathbf{y}}$ are their counterparts for noisy speech.

4 Comparison of Approximation Methods by Simulation

In order to compare the accuracies of the different approximation methods proposed in this paper as well as the ones in literature that include the *log-normal approximation* in [2] and the *first-order VTS approximation* in [1,7,8], a series of simulation experiments are conducted. For simplicity, we consider three random variables: y, x, n, corresponding to the 1-dimensional case described in Eq. (1). Suppose the means and standard deviations of x and n are known. We want to

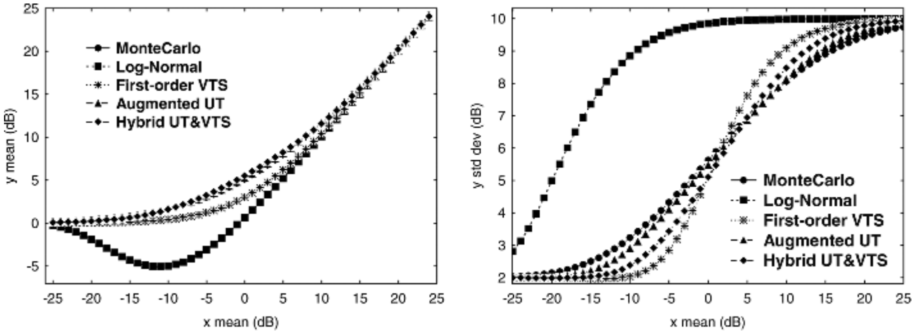


Fig. 1. A comparison of means and standard deviations of y in Eq. (1) estimated by different methods: Monte Carlo simulation, log-normal approximation, first-order VTS, augmented UT, and hybrid UT&VTS approximation ($\mu_{\mathbf{n}} = 0dB$, $\sigma_{\mathbf{n}} = 2dB$, $\sigma_{\mathbf{x}} = 10dB$, and $\mu_{\mathbf{x}}$ varies from $-25dB$ to $25dB$)

estimate the mean and standard deviation of y by using different approximation methods, and compare their accuracies with that of using Monte Carlo simulation. In Fig. 1, we compare the estimates (in dB) of the mean and standard deviation of y obtained by the abovementioned approximation methods as a function of the μ_x (in dB), when $\sigma_x = 10dB$, $\mu_n = 0dB$ and $\sigma_n = 2dB$. It is observed that the Augmented UT approach (i.e. Method 1 described in Section 2.2) offers the most accurate estimations among the four approximation methods for both the mean and standard deviation, which are actually very close to the results of the Monte Carlo simulation. The hybrid UT&VTS approach (i.e. Method 2 described in Section 2.3) provides a good estimation for the mean and a better estimation for the standard deviation than the first-order VTS method does. Among the four approximation methods, log-normal approximation performs the worst.

5 Experiments and Results

5.1 Experimental Setup

In order to verify the effectiveness of the proposed UT approaches and compare them to the log-normal-approximation-based PMC [2] and the first-order-approximation-based VTS approaches [1,7,8], a series of experiments are performed for the task of speaker independent recognition of connected digit strings on Aurora2 database. A full description of the Aurora2 database and a test framework is given in [3].

Our CDHMM-based ASR system is trained from the "clean" speech data in Aurora2 database and a 39-dimensional feature vector is used, which consists of 13 MFCCs (including C_0) plus their first and second order derivatives. The speech data is processed in a time window of 25ms, shifted every 10ms. A pre-emphasis with a coefficient of 0.97 is performed. The number of Mel-frequency

Table 1. Performance (word accuracy in %) comparison of different HMM compensation methods averaged over SNRs between 0 and 20 dB on three different test sets of Aurora2 database (R.E.R. stands for the relative error rate reduction in % *vs.* the baseline performance without HMM compensation)

Methods	**Set A**	**Set B**	**Set C**	**Overall**	**R.E.R.**
Baseline	57.67	54.41	64.89	57.81	N/A
Log-Normal PMC	84.56	83.45	84.97	84.20	62.54
First-Order VTS	84.77	83.56	84.82	84.30	62.78
Augmented UT	86.01	85.02	86.83	85.78	66.29
Hybrid UT&VTS	85.82	84.82	86.32	85.52	65.67
Simplex UT	86.20	85.05	86.39	85.78	66.29

filter banks is 23. The delta and delta-delta features are extracted using linear regression method as detailed in [9] with a setting of relevant parameters in HTK notations as $deltawindow = 3$ and $accwindow = 2$.

Both training and recognition were performed by using the HTK [9] and the standard scripts provided by ETSI [3]. The mixture number of each CDHMM state is 3. The single Gaussian model of additive noise in each test sentence is estimated from noise frames at the beginning and end of the sentence in cepstral domain. Because we focus our study in this paper on HMM compensation for additive noise only, no other compensation is performed to cope with other possible distortions. The relevant static parameters of CDHMMs are compensated by using the log-normal-approximation-based PMC, the first-order-approximation-based VTS, and our proposed UT approaches, but dynamic parameters are all compensated by the same method described in Eq. (28). The relevant control parameters in our UT approaches are set as $\kappa^a = 0$ in Eq. (9) and Eq. (13), $\kappa = 0$ in Eq. (18), and $W_0 = 0$ in the first step of the Method 3 in Section 2.4.

5.2 Experimental Results

Table 1 summarizes a performance (word accuracy in %) comparison of different HMM compensation methods, where the performance is averaged over SNRs between 0 and 20 dB on each of the three different test sets of Aurora2 database, namely Set A, Set B and Set C. It is observed that all of our proposed UT-based methods achieve a better performance in all the test sets than that of both the PMC and VTS methods. There is no big performance difference among three UT-based methods. This may suggest that the most computationally efficient simplex UT based method can be used in practice for HMM compensation if the computation time is a concern.

Table 2 provides a performance (word accuracy in %) comparison of different HMM compensation methods averaged over three test sets of Aurora2 database at each SNR (in dB). Again, it is observed that the UT methods perform better than both the PMC and VTS methods at different SNRs.

Table 3 summarizes the relative error rate reductions (in %) of the augmented UT method *vs.* the first-order VTS method under different combinations of SNR

Table 2. Performance (word accuracy in %) comparison of different HMM compensation methods averaged over three test sets of Aurora2 database at each SNR

Methods	**0 dB**	**5 dB**	**10 dB**	**15 dB**	**20 dB**	**Overall**
Baseline	14.19	33.19	62.50	84.64	94.55	57.81
Log-Normal PMC	59.19	79.61	90.02	94.85	97.31	84.20
First-Order VTS	61.21	80.14	89.66	93.98	96.51	84.30
Augmented UT	62.25	82.35	91.56	95.38	97.35	85.78
Hybrid UT&VTS	62.15	81.79	91.21	95.16	97.26	85.52
Simplex UT	62.69	82.40	91.38	95.22	97.20	85.78

Table 3. Summary of relative error rate reductions (in %) of the augmented UT method *vs.* the first-order VTS method under different combinations of SNR and noise type of the test Set A on Aurora2 database

Noise Conditions	**Subway**	**Babble**	**Car**	**Exhibition**	**Average**
20 dB	29.46	23.10	0.00	4.15	14.18
15 dB	21.28	22.04	9.97	7.08	15.09
10 dB	24.03	17.10	8.27	9.75	14.79
5 dB	16.82	7.33	6.23	10.75	10.28
0 dB	8.26	1.74	-2.94	6.46	3.38
Average	14.55	7.83	1.31	7.93	8.15

and noise type of the test Set A on Aurora2 database. It is observed that the performance difference between two methods in car noise environment is not as big as in other noise environments. A similar observation is also made for train station noise environment in test Set B not shown here. After a detailed analysis of the noise distribution statistics in each noise environment, we noticed that there are big differences of the noise variances. For example, at $SNR = 10dB$, the average of the standard deviations in all dimensions of noise Gaussian models in the log-spectral domain are 0.126, 0.140, 0.082, 0.092 for four types of noises in the test Set A, namely *Subway*, *Babble*, *Car*, *Exhibition*, respectively. It seems that when the noise variances are small, the negative effects by ignoring the truncated higher-order terms in VTS expansion are smaller, therefore the VTS method will perform relatively better in those cases. Overall, the UT method performs much better than the VTS method.

6 Summary and Future Works

In this paper, we have introduced a new HMM compensation approach using a technique called Unscented Transformation (UT). As a first step, we have studied three implementations of the UT approach with different computational complexities for noisy speech recognition, and evaluated their performance on Aurora2 connected digits database. It is demonstrated that the UT approaches achieve significant improvements in recognition accuracy compared to the log-normal-approximation-based PMC and first-order-approximation-based VTS approaches.

Ongoing and future works include 1) to extend UT methods for HMM compensation to cope with both convolutional and additive distortions, 2) to apply UT-based techniques for feature compensation, 3) to develop UT-based techniques for dealing with nonstationary distortions, 4) to evaluate the above techniques on different tasks and databases. We will report those results elsewhere when they become available.

References

1. A. Acero, L. Deng, T. Kristjansson, and J. Zhang, “HMM adaptation using vector Taylor series for noisy speech recognition,” in *Proc. ICSLP*, Beijing, 2000, pp. 869-872.
2. M. J. F. Gales, *Model-based Techniques For Noise Robust Speech Recognition*, Ph.D. thesis, Cambridge University, UK, 1995.
3. H. G. Hirsch and D. Pearce, “The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions,” in *Proc. ISCA ITRW ASR*, Paris, France, Sep. 2000, pp.181-188.
4. S. J. Julier, “The spherical simplex unscented transformation,” in *Proc. Amer. Control Conf.*, Denver, Colorado, June 2003, pp. 2430-2434.
5. S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, Vol. 92, No. 3, pp.401-422, 2004.
6. D.-Y. Kim, C.-K. Un, and N.-S. Kim, “Speech recognition in noisy environments using first-order vector Taylor series,” *Speech Communication*, Vol. 24, pp.39-49, 1998.
7. P. J. Moreno, *Speech Recognition in Noisy Environments*, Ph.D. thesis, Carnegie Mellon University, 1996.
8. P. J. Moreno, B. Raj, and R. M. Stern, “A vector Taylor series approach for environment-independent speech recognition,” in *Proc. ICASSP*, Atlanta, 1996, pp.733-736.
9. S. J. Young, *et al.*, *The HTK Book* (revised for HTK Version 3.3), 2005.

Noisy Speech Recognition Performance of Discriminative HMMs

Jun Du[1], Peng Liu[2], Frank Soong[2], Jian-Lai Zhou[2], and Ren-Hua Wang[1]

[1] University of Science and Technology of China, Hefei, 230027
[2] Microsoft Research Asia, Beijing, 100080
unuedjwj@ustc.edu, {pengliu, frankkps, jlzhou}@microsoft.com,
rhw@ustc.edu.cn

Abstract. Discriminatively trained HMMs are investigated in both clean and noisy environments in this study. First, a recognition error is defined at different levels including string, word, phone and acoustics. A high resolution error measure in terms of minimum divergence (MD) is specifically proposed and investigated along with other error measures. Using two speaker-independent continuous digit databases, Aurora2(English) and CNDigits (Mandarin Chinese), the recognition performance of recognizers, which are trained in terms of different error measures and using different training modes, is evaluated under different noise and SNR conditions. Experimental results show that discriminatively trained models performed better than the maximum likelihood baseline systems. Specifically, for MD trained systems, relative error reductions of 17.62% and 18.52% were obtained applying multi-training on Aurora2 and CNDigits, respectively.

Keywords: Noise Robustness, Minimum Divergence, Minimum Word Error, Discriminative Training.

1 Introduction

With the progress of Automatic Speech Recognition (ASR), noise robustness of speech recognizers attract more and more attentions for practical recognition systems. Various noise robust technologies which can be grouped into three classes. 1. Feature domain approaches, which aim at noise resistant features, e.g., speech enhancement, feature compensation or transformation methods [1]; 2. Model domain approaches, e.g., Hidden Markov Model (HMM) decompensation [2], Parallel Model Combination (PMC) [3], which aim at modeling the distortion of features in noisy environments directly; 3. Hybrid approaches.

In the past decade, discriminative training has been shown quite effective in reducing word error rates of HMM based ASR systems in clean environment. In the first stage, sentence level discriminative criteria, including Maximum Mutual Information (MMI) [4,5], Minimum Classification Error (MCE) [6], were proposed and proven effective. Recently, new criteria such as Minimum Word Error (MWE) and Minimum Phone Error (MPE) [7], which are based on fine

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 358–369, 2006.

error analysis at word or phone level, have achieved further improvement in recognition performance.

In [8,9,10], noise robustness investigation on sentence level discriminative criteria such as MCE, Corrective Training (CT) is reported. Hence, we are motivated to give a more complete investigation of noise robustness for genaral minimum error training.

From a unified viewpoint of error minimization, MCE, MWE and MPE are only different in error definition. String based MCE is based upon minimizing sentence error rate, while MWE is based on word error rate, which is more consistent with the popular metric used in evaluating ASR systems. Hence, the latter yields better word error rate, at least on the training set [7]. However, MPE performs slightly but universally better than MWE on testing set [7]. The success of MPE might be explained as follows: when refining acoustic models in discriminative training, it makes more sense to define errors in a more granular form of acoustic similarity. However, binary decision at phone label level is only a rough approximation of acoustic similarity.

Therefore, we propose to use acoustic dissimilarity to measure errors. Because acoustic behavior of speech units are characterized by HMMs, by measuring Kullback-Leibler Divergence (KLD) [11] between two given HMMs, we can have a physically more meaningful assessment of their acoustic similarity. Given sufficient training data, "ideal" ML models can be trained to represent the underlying distributions and then can be used for calculating KLDs.

Adopting KLD for defining errors, the corresponding training criterion is referred as Minimum Divergence (MD) [12]. The criterion possesses the following advantages: 1) It employs acoustic similarity for high-resolution error definition, which is directly related with acoustic model refinement; 2) Label comparison is no longer used, which alleviates the influence of chosen language model and phone set and the resultant hard binary decisions caused by label matching. Because of these advantages, MD is expected to be more flexible and robust.

In our work, MWE, which matches the evaluation metric, and MD, which focus on refining acoustic dissimilarity, are compared. Other issues related to robust discriminative training, including how to design the maximum likelihood baseline, and how to treat with silence model is also discussed.

Experiments were performed on Aurora2 [13], which is a widely adopted database for research on noise robustness, and CNDigits, a Chinese continuous digit database. We tested the effectiveness of discriminative training on different ML baseline and different noise environments.

The rest of paper is organized as follows. In section 2, issues on noise robustness of minimum error training will be discussed. In section 3, MD training will be introduced. Experimental results are shown and discussed in section 4. Finally in section 5, we give our conclusions.

2 Noise Robustness Analysis of Minimum Error Training

In this section, we will have a general discuss on the major issues we are facing in robust discriminative training.

2.1 Error Resolution of Minimum Error Training

In [7] and [12], various discriminative trainings in terms of their corresponding optimization measures are unified under the framework of minimum error training, where the objective function is an average of the recognition accuracies $\mathcal{A}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}})$ of all hypotheses weighted by the posterior probabilities. For conciseness, we consider single utterance case:

$$\mathcal{F}(\boldsymbol{\theta}) = \sum_{\boldsymbol{W} \in \boldsymbol{\mathcal{M}}} P_{\boldsymbol{\theta}}(\boldsymbol{W} \mid \boldsymbol{O}) \mathcal{A}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}}) \tag{1}$$

where $\boldsymbol{\theta}$ represents the set of the model parameters; $\boldsymbol{O}$ is a sequence of acoustic observation vectors; $\boldsymbol{W}_{\mathrm{r}}$ is the reference word sequence; $\boldsymbol{\mathcal{M}}$ is the hypotheses space; $P_{\boldsymbol{\theta}}(\boldsymbol{W} \mid \boldsymbol{O})$ is the generalized posterior probability of the hypothesis $\boldsymbol{W}$ given $\boldsymbol{O}$, which can be formulated as:

$$P_{\boldsymbol{\theta}}(\boldsymbol{W} \mid \boldsymbol{O}) = \frac{P_{\boldsymbol{\theta}}^{\kappa}(\boldsymbol{O} \mid \boldsymbol{W}) P(\boldsymbol{W})}{\sum_{\boldsymbol{W}' \in \boldsymbol{\mathcal{M}}} P_{\boldsymbol{\theta}}^{\kappa}(\boldsymbol{O} \mid \boldsymbol{W}') P(\boldsymbol{W}')} \tag{2}$$

where κ is the acoustic scaling factor.

The gain function $\mathcal{A}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}})$ is an *accuracy* measure of $\boldsymbol{W}$ given its reference $\boldsymbol{W}_{\mathrm{r}}$. In Table 1, comparison among several minimum error criteria are tabulated. In MWE training, $\mathcal{A}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}})$ is word accuracy, which matches the commonly used evaluation metric of speech recognition. However, MPE has been shown to be more effective in reducing recognition errors because it provides a more precise measurement of word errors at the phone level. We can argue this point by advocating the final goal of discriminative training. In refining acoustic models to obtain better performance, it makes more sense to measure acoustic similarity between hypotheses instead of word accuracy. The symbol matching does not relate acoustic similarity with recognition. The measured errors can also be strongly affected by the phone set definition and language model selection. Therefore, acoustic similarity is proposed as a finer and more direct error definition in MD training.

Here we aim to seeking how criteria with different error resolution performs in the noisy environments. In our experiments, whole-word model, which is commonly used in digit tasks, is adopted. For the noisy robustness analysis, MWE which

Table 1. Comparison among criteria of minimum error training. ($\boldsymbol{P}_{\boldsymbol{W}}$: *Phone sequence corresponding to word sequence* $\boldsymbol{W}$; LEV(,): *Levenshtein distance between two symbol strings*; $|\cdot|$: *Number of symbols in a string.*)

Criterion	$\mathcal{A}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}})$	Objective
String based MCE	$\delta(\boldsymbol{W} = \boldsymbol{W}_{\mathrm{r}})$	Sentence accuracy
MWE	$\lvert\boldsymbol{W}_{\mathrm{r}}\rvert - \mathrm{LEV}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}})$	Word accuracy
MPE	$\lvert P_{\boldsymbol{W}_{\mathrm{r}}}\rvert - \mathrm{LEV}(P_{\boldsymbol{W}}, P_{\boldsymbol{W}_{\mathrm{r}}})$	Phone accuracy
MD	$-D(\boldsymbol{W}_{\mathrm{r}} \parallel \boldsymbol{W})$	Acoustic similarity

matches with the model type and evaluation metric of speech recognition, will compared with MD, which possesses the highest error resolution as shown in Table 1.

2.2 Different Training Modes

In noisy environments, various ML trained baseline can be designed. So the effectiveness of minimum error training with different training modes will be explored. In [13], two different sets of training, clean-training and multi-training, are used. In clean-training mode, only clean speech is used for training. Hence, when testing in noisy environments, there will be a mismatch. To alleviate this mismatch, multi-training, in which training set is composed of noisy speech with different SNRs, can be applied. But multi-training can only achieve a "global SNR" match. To achieve a "local SNR" match, we propose a SNR-based training mode. In our SNR-based training, each HMM set is trained using the speech with a specific SNR. A big HMM set is composed of all the SNR-based HMM sets. So there will be several SNR-based models for each digit. When testing, we will adopt the multi-pronunciation dictionary to output the digital label. SNR-based training can be considered as a high resolution acoustic modeling of multi-training. Illustration of three training modes is shown in Fig. 1.

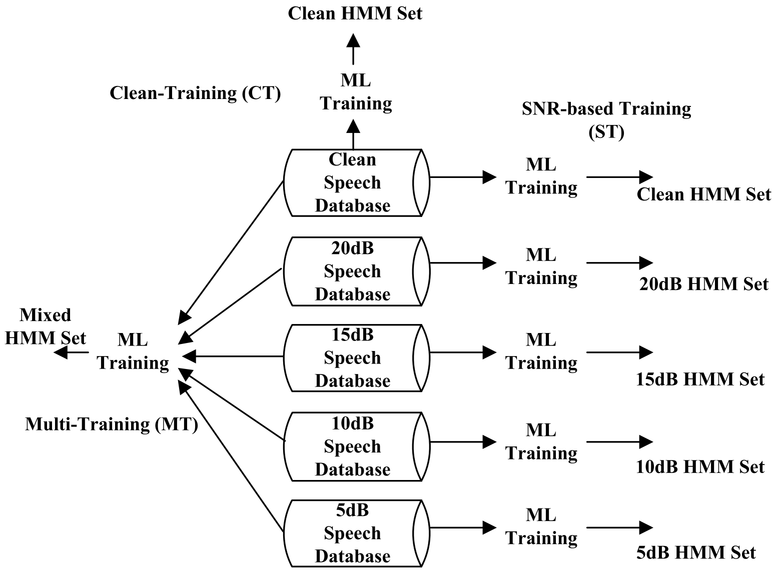


Fig. 1. Illustration of three training modes

2.3 Silence Model Update

Silence or background model can have a significant effect on word errors. Hence, whether or not to update silence model in minimum error training can be critical under noisy conditions. In our research, we pay special attention to this issue for reasonable guidelines.

3 Word Graph Based Minimum Divergence Training

3.1 Defining Errors by Acoustic Similarity

A word sequence is acoustically characterized by a sequence of HMMs. For automatically measuring acoustic similarity between $\boldsymbol{W}$ and $\boldsymbol{W}_{\mathrm{r}}$, we adopt KLD between the corresponding HMMs:

$$\mathcal{A}(\boldsymbol{W}, \boldsymbol{W}_{\mathrm{r}}) = -D(\boldsymbol{W}_{\mathrm{r}} \parallel \boldsymbol{W}) \tag{3}$$

The HMMs, when they are reasonably well trained in ML sense, can serve as succinct descriptions of data.

3.2 KLD Between Two Word Sequences

Given two word sequences $\boldsymbol{W}_{\mathrm{r}}$ and $\boldsymbol{W}$ without their state segmentations, we should use a state matching algorithm to measure the KLD between the corresponding HMMs [14]. With state segmentations, the calculation can be further decomposed down to the state level:

$$\begin{aligned} D(\boldsymbol{W}_{\mathrm{r}} \parallel \boldsymbol{W}) &= D(\boldsymbol{s}_{\mathrm{r}}^{1:T} \parallel \boldsymbol{s}^{1:T}) \\ &= \int p(\boldsymbol{o}^{1:T} \,|\, \boldsymbol{s}_{\mathrm{r}}^{1:T}) \log \frac{p(\boldsymbol{o}^{1:T} \,|\, \boldsymbol{s}_{\mathrm{r}}^{1:T})}{p(\boldsymbol{o}^{1:T} \,|\, \boldsymbol{s}^{1:T})} d\boldsymbol{o}^{1:T} \end{aligned} \tag{4}$$

where T is the number of frames; $\boldsymbol{o}^{1:T}$ and $\boldsymbol{s}_{\mathrm{r}}^{1:T}$ are the observation sequence and hidden state sequence, respectively.

By assuring all observations are independent, we obtain:

$$D(\boldsymbol{s}_{\mathrm{r}}^{1:T} \parallel \boldsymbol{s}^{1:T}) = \sum_{t=1}^{T} D(s_{\mathrm{r}}^{t} \parallel s^{t}) \tag{5}$$

which means we can calculate KLD state by state, and sum them up.

Conventionally, each state s is characterized by a Gaussian Mixture Model (GMM): $p(\boldsymbol{o} \,|\, s) = \sum_{m=1}^{M_s} \omega_{sm} \mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}_{\boldsymbol{sm}}, \boldsymbol{\Sigma}_{\boldsymbol{sm}})$, so the comparison is reduced to measuring KLD between two GMMs. Since there is no closed-form solution, we need to resort to the computationally intensive Monte-Carlo simulations. The unscented transform mechanism [15] has been proposed to approximate the KLD measurement of two GMMs.

Let $\mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ be a N-dimensional Gaussian distribution and h is an arbitrary $\mathbb{R}^N \rightarrow \mathbb{R}$ function, unscented transform mechanism suggests approximating the expectation of h by:

$$\int \mathcal{N}(\boldsymbol{o}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) h(\boldsymbol{o}) d\boldsymbol{o} \approx \frac{1}{2N} \sum_{k=1}^{2N} h(\boldsymbol{o}_k) \tag{6}$$

where $\boldsymbol{o}_k(1 \le k \le 2N)$ are the artificially chosen "*sigma*" points: $\boldsymbol{o}_k = \boldsymbol{\mu} + \sqrt{N\lambda_k}\boldsymbol{u}_k$, $\boldsymbol{o}_{k+N} = \boldsymbol{\mu} - \sqrt{N\lambda_k}\boldsymbol{u}_k(1 \le k \le N)$, where λ_k, $\boldsymbol{u}_k$ are the k^{th} eigenvalue and eigenvector of $\boldsymbol{\Sigma}$, respectively. Geometrically, all these "*sigma*" points are on the principal axes of $\boldsymbol{\Sigma}$. Eq. 6 is precise if h is quadratic.

Based on Eq. 6, KLD between two Gaussian mixtures is approximated by:

$$D(s_{\mathrm{r}} \,\|\, s) \approx \frac{1}{2N} \sum_{m=1}^{M} \omega_m \sum_{k=1}^{2N} \log \frac{p(\boldsymbol{o}_{m,k} \,|\, s_{\mathrm{r}})}{p(\boldsymbol{o}_{m,k} \,|\, s)} \tag{7}$$

where $\boldsymbol{o}_{m,k}$ is the k^{th} "*sigma*" point in the m^{th} Gaussian kernel of $p(\boldsymbol{o}_{m,k} \,|\, s_{\mathrm{r}})$. By plugging it into Eq. 4, we obtain the KLD between two word sequences given their state segmentations.

3.3 Gain Function Calculation

Usually, word graph is a compact representation of large hypotheses space in speech recognition. Because the KLD between a hypothesised word sequence and the reference can be decomposed down to the frame level, we have the following word graph based representation of (1):

$$\mathcal{F}(\boldsymbol{\theta}) = \sum_{w \in \boldsymbol{\mathcal{M}}} \sum_{\boldsymbol{W} \in \boldsymbol{\mathcal{M}}: w \in \boldsymbol{W}} P_{\boldsymbol{\theta}}(\boldsymbol{W} \,|\, \boldsymbol{O}) \mathcal{A}(w) \tag{8}$$

where $\mathcal{A}(w)$ is the gain function of word arc w. Denote b_w, e_w the start frame index and end frame index of w, we have:

$$\mathcal{A}(w) = -\sum_{t=b_w}^{e_w} D(s_w^t \,\|\, s_{\mathrm{r}}^t) \tag{9}$$

where the s_w^t and s_{r}^t represent the certain state at time t on arc w and the reference, respectively.

As mentioned in [7], we use Forward-Backward algorithm to update the word graph and the Extended Baum-Welch algorithm to update the model parameters in the training iterations.

4 Experiments

4.1 Experimental Setup

Experiments on both English (TIDigits and Aurora2) and Chinese (CNDigits) continuous digit tasks were performed. The English vocabulary is made of the 11 digits, from 'one(1)' to 'nine(9)', plus 'oh(0)' and 'zero(0)'. The Chinese vocabulary is made of digits from 'ling(0)' to 'jiu(9)', plus 'yao(1)'. The baseline configuration for three systems is listed in Table 2.

For TIDigits Experiments, man, woman, boy and girl speakers, were used in both training and testing.

The Aurora2 task consists of English digits in the presence of additive noise and linear convolutional channel distortion. These distortions have been synthetically introduced to clean TIDigits data. Three testing sets measure performance against noise types similar to those seen in the training data (set A), different from those seen in the training data (set B), and with an additional convolutional channel (set C). The baseline performance and other details can be found in [13].

The original clean database of CNDigits is collected by Microsoft Research Asia. 8 types of noises, i.e. waiting room of a station, platform, shop, street, bus, airport lounge, airport exit, outside, are used for noise addition. 8000 clean utterances from 120 female and 200 male speakers for training set are split into 20 subsets with 400 utterances in each subset. Each subset contains a few utterances of all training speakers. The 20 subsets represent 4 different noise scenarios at 5 different SNRs. The 4 noises are waiting room, street, bus and airport lounge. The SNRs are 20dB, 15dB, 10dB, 5dB and the clean condition. Two different test sets are defined. 3947 clean utterances from 56 female and 102 male speakers are split into 4 subsets with about 987 utterances in each. All speakers are involved in each subset. One noise is added to each subset at SNRs of 20dB, 15dB, 10dB, 5dB, 0dB, -5dB and the clean condition. In the first test set, called test set WM(Well-Match), the four noises, the same as those used in training set, are added to the 4 subset. The second test set, called test set MM(Mis-Match), is created in exactly the same way, but using four different noises , namely platform, shop, airport exit and outside. Our design of CNDigits database is similar to Aurora2.

For mininum error training, the acoustic scaling factor κ was set to $\frac{1}{33}$. All KLDs between any two states were precomputed to make the MD training more efficient. For Aurora2 and CNDigits, we select the best results after 20 iterations for each sub set of testing.

Table 2. Baseline configuration

System	Feature	Model Type	# State /Digit	# Gauss /State	# string of training set	# string of testing set
TIDigits	MFCC_E_D_A	left-to-right whole-word model without skipping	10	6	12549	12547
Aurora2			16	3	8440*2	1001*70
CNDigits			10	3	8000	987*56

4.2 Experiments on TIDigits Database

As a preliminary of noise robustness analysis, we first give the results of MD on the clean TIDigits database compared with MWE. As shown in Fig. 2,

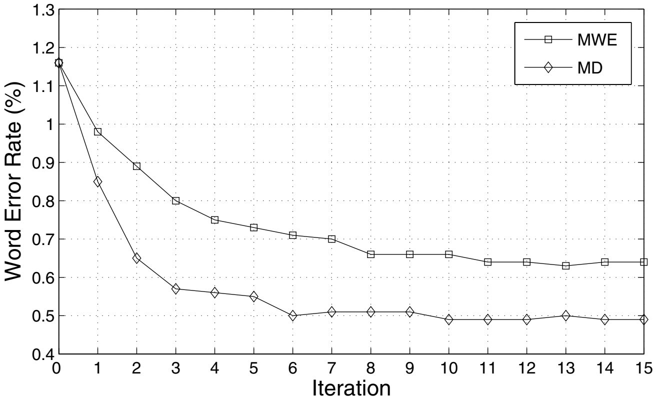


Fig. 2. Performance comparison on TIDigits

performance of MD achieves 57.8% relative error reduction compared with ML baseline and also outperforms MWE in all iterations.

4.3 Experiments on Aurora2 Database

Silence Model Update. As shown in Table 3, we explore whether to update silence model in minimum error training using different training modes. Because it is unrelated with criteria, here we adopt MWE. when applying clean-training, the performances on all test sets without updating silence model are consistently better. But in multi-training, the conclusion is opposite. From the results, we can conclude that increasing the discrimination of silence model will lead to performance degradation in mismatched cases (clean-training) and performance improvement in matched cases (multi-training). Obviously our SNR-based training belongs to the latter. In all our experiments, the treatment of silence model will obey this conclusion.

Table 3. Word Accuracy (%) of MWE with or without silence model update in different training modes on Aurora2

Training Mode	Update Silence Model	Set A	Set B	Set C	Overall
Clean	YES	61.85	56.94	66.26	60.77
Clean	NO	64.74	61.69	67.95	64.16
Multi	YES	89.15	89.16	84.66	88.26
Multi	NO	88.91	88.55	84.43	87.87

Error Resolution of Minimum Error Training. As shown in Table 4, the performances of MD and MWE are compared. Here multi-training is adopted because it's believed that matching between training and testing can tap the potential of minimum error training. For the overall performance on three test sets, MD consistently outperforms MWE. From the viewpoint of SNRs, MD outperforms MWE in most cases when SNR is below 15dB. Hence, we can conclude that although MWE matches with the model type and evaluation metric of speech recognition, MD which possesses the highest error resolution outperforms it in low SNR. In other words, the performance can be improved in low SNR by increasing the error resolution of criterion in minimum error training.

Table 4. Performance comparison on Aurora2 (MD vs. MWE)

Multi-Training - Results (Minimum Divergence)															
	A					B					C				Rel
	Subway	Babble	Car	Exhibitior	Average	Restaurant	Street	Airport	Station	Average	Subway M	Street M	Average	Average	Impr
Clean	99.14	99.12	98.9	99.2	99.09	99.14	99.12	98.9	99.2	99.09	98.89	98.85	98.87	99.05	35.32%
20 dB	98.71	98.55	98.81	98.61	98.67	98.43	98.37	98.57	98.89	98.57	98.65	97.64	98.15	98.52	43.92%
15 dB	98.5	98	98.33	97.93	98.19	98	97.76	97.79	97.93	97.87	97.88	96.74	97.31	97.89	42.04%
10 dB	97.18	96.55	97.2	96.08	96.75	96.41	95.8	96.06	95.31	95.90	95.15	94.04	94.60	95.98	34.81%
5 dB	92.39	89.81	90.49	90.25	90.74	89.28	87.06	90.52	87.23	88.52	84.68	82.56	83.62	88.43	20.78%
0 dB	72.8	64.63	58.93	70.32	66.67	65.24	64	69.19	62.48	65.23	49.25	54.44	51.85	63.13	10.51%
-5dB	31.04	29.56	22.7	28.57	27.97	30.06	28.96	33.58	25.46	29.52	22.01	24.24	23.13	27.62	4.15%
Average	91.92	89.51	88.75	90.64	90.20	89.47	88.60	90.43	88.37	89.22	85.12	85.08	85.10	88.79	
Rel Impr	28.10%	12.93%	16.53%	21.79%	19.60%	27.93%	12.04%	22.53%	22.40%	21.45%	11.21%	4.93%	8.17%		17.62%

Multi-Training - Results (Minimum Word Error)															
	A					B					C				Rel
	Subway	Babble	Car	Exhibitior	Average	Restaurant	Street	Airport	Station	Average	Subway M	Street M	Average	Average	Impr
Clean	99.14	99.18	99.02	99.29	99.16	99.14	99.18	99.02	99.29	99.16	98.99	99.06	99.03	99.13	40.96%
20 dB	98.86	98.67	98.78	98.7	98.75	98.74	98.43	98.72	98.95	98.71	98.34	97.4	97.87	98.56	45.45%
15 dB	98.74	98.13	98.33	97.69	98.22	98.5	97.82	98.03	98.06	98.10	97.33	96.25	96.79	97.89	41.97%
10 dB	96.87	95.95	96.87	95.43	96.28	96.22	95.53	96.42	95.74	95.98	94.63	93.5	94.07	95.72	30.03%
5 dB	92.32	88.85	88.25	88.83	89.56	88.36	87.3	89.53	86.61	87.95	84.49	82.62	83.56	87.72	15.40%
0 dB	70.31	63.33	53.44	64.7	62.95	64.6	68.18	68.27	59.12	65.04	47.62	54.44	51.03	61.40	6.25%
-5dB	29.66	29.72	21.8	25.27	26.61	30.21	27.84	33.49	23.97	28.88	21.31	24.24	22.78	26.75	3.01%
Average	91.42	88.99	87.13	89.07	89.15	89.28	89.45	90.19	87.70	89.16	84.48	84.84	84.66	88.26	
Rel Impr	23.69%	8.60%	4.53%	8.69%	10.98%	26.64%	18.62%	20.65%	17.92%	21.02%	7.39%	3.39%	5.46%		13.71%

Different Training Modes. Fig. 3 shows relative improvement over ML baseline using MD training with different training modes. From this figure, some conclusions can be obtained. First, Set B, whose noise scenarios are different from training achieves the most obvious relative improvement in most cases. The relative improvement of set A are comparable with set B in the clean-training and multi-training but worse than set B in SNR-based training. The relative improvement of set C, due to the mismatch of noise scenario and channel, almost the worst in all training modes. Second, the relative improvement performance declines for decreasing SNR in clean-training. But in multi-training and SNR-based training, the peak performance is in the range of 20dB to 15dB. Also in the low SNRs, the performance of cleaning-training is worse than the other two training modes on set A and set B.

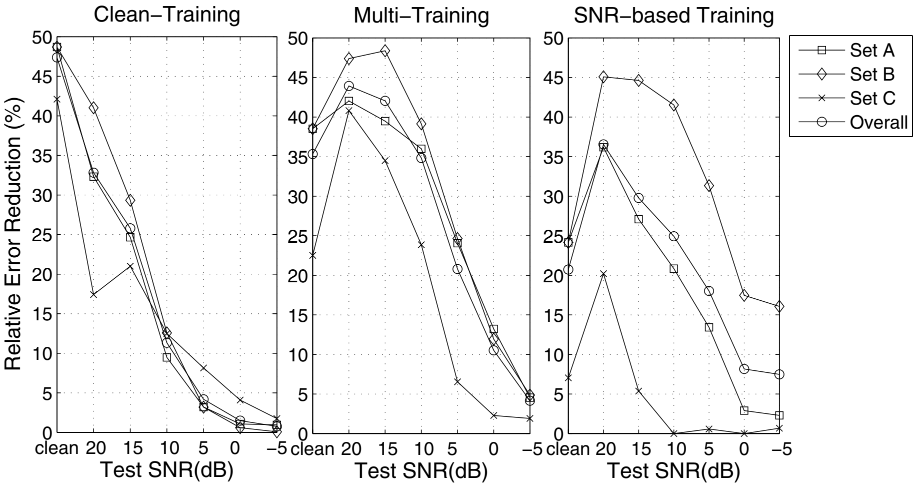


Fig. 3. Relative Improvement over ML baseline on Aurora2 using different training modes in MD training

Table 5. Summary of performance on Aurora2 using different training modes in MD training

	Word Accuracy (%)	Relative Improvement
Training Mode	Set A Set B Set C Overall	Set A Set B Set C Overall
Clean-Training	63.49 58.94 68.96 62.76	5.56% 7.21% 8.32% 6.76%
Multi-Training	90.20 89.22 85.10 88.79	19.60% 21.45% 8.17% 17.62%
SNR-based Training	91.27 89.27 86.70 89.56	10.00% 26.21% 1.14% 15.68%

The summary of performance is listed in Table 5. Word accuracy of our SNR-based training outperforms multi-training on all test sets, especially set A and set C. For the overall relative improvement, the best result of 17.62% is achieved in multi-training.

4.4 CNDigits Database Experiments

On CNDigits database, we compare MD and MWE with ML applying multi-training as a further verification of conclusions on Aurora2. Performances are shown in Table 6. Totally MD achieves 18.52% relative improvement over ML baseline. Although minimum error training on both English and Chinese is effective in noisy envrionments, there are still some differences. First, the most obvious relative improvement on CNDigits occurs in clean condition which is different from that on Aurora2. Second, more than 10% relative improvement is still obtained at low SNRs (below 0dB) on CNDigits. Third, MD outperforms MWE in all noisy conditions.

Table 6. Performance comparison on Chinese digit database (CNDigits) using multi-training

Multi-Training - Results (ML Reference)											
	Well-Matched(WM)					Mis-Matched(MM)					
	Waiting Room	Street	Bus	Lounge	Average	Platform	Shop	Outside	Exit	Average	Average
Clean	95.9	95.46	95.57	95.33	95.57	95.9	95.46	95.57	95.33	95.57	95.57
20 dB	95.54	95.35	95.56	95.07	95.38	95.93	94.99	95.55	95.17	95.41	95.40
15 dB	94.21	95.29	95.61	94.63	94.94	95.45	93.83	95.33	94.88	94.87	94.90
10 dB	91.15	94.17	95.52	93.82	93.67	94.12	90.4	94.2	94.25	93.24	93.45
5 dB	82.33	92.21	95.42	89.64	89.90	89.64	80.97	90.63	91.34	88.15	89.02
0 dB	65.42	84.63	94.85	77.43	80.58	77.4	64.46	80.77	82.36	76.25	78.42
-5dB	39.23	68.18	93.34	51.1	62.96	51.3	39.53	57.78	58.7	51.83	57.40
Average	85.73	92.33	95.39	90.12	90.89	90.51	84.93	91.30	91.60	89.58	90.24

Multi-Training - Results (Minimum Word Error)												
	Well-Matched(WM)					Mis-Matched(MM)						Rel
	Waiting Room	Street	Bus	Lounge	Average	Platform	Shop	Outside	Exit	Average	Average	Impr
Clean	96.95	96.45	96.74	96.52	96.67	96.95	96.45	96.74	96.52	96.67	96.67	24.83%
20 dB	96.19	96.33	96.74	96.26	96.38	96.73	95.96	96.66	96.3	96.41	96.40	21.72%
15 dB	94.84	96.06	96.78	95.76	95.86	95.95	94.69	96.12	95.96	95.68	95.77	17.23%
10 dB	92.32	94.84	96.72	94.39	94.57	94.53	91.62	94.99	94.73	93.97	94.27	12.80%
5 dB	86.09	92.66	96.47	90.53	91.44	90.07	84.5	91.91	92.05	89.63	90.54	12.89%
0 dB	70.89	85.06	95.99	78.82	82.69	78.72	67.74	82.18	83.47	78.03	80.36	9.45%
-5dB	42.58	69.2	94.33	51.89	64.50	51.72	40.46	58.99	59.48	52.66	58.58	4.04%
Average	88.07	92.99	96.54	91.15	92.19	91.20	86.90	92.37	92.50	90.74	91.47	
Rel Impr	16.37%	8.60%	24.91%	10.46%	14.21%	7.29%	13.09%	12.36%	10.74%	11.14%		12.57%

Multi-Training - Results (Minimum Divergence)												
	Well-Matched(WM)					Mis-Matched(MM)						Rel
	Waiting Room	Street	Bus	Lounge	Average	Platform	Shop	Outside	Exit	Average	Average	Impr
Clean	97.21	96.67	97.06	96.59	96.88	97.21	96.67	97.06	96.59	96.88	96.88	29.80%
20 dB	96.19	96.46	96.98	96.29	96.48	96.92	96.1	96.84	96.38	96.56	96.52	24.46%
15 dB	95.02	96.35	97.08	95.97	96.11	96.26	94.77	96.44	96.11	95.90	96.00	21.97%
10 dB	92.44	95.02	96.87	94.64	94.74	94.78	91.93	95.29	94.92	94.23	94.49	16.27%
5 dB	86.47	92.75	96.77	90.87	91.72	90.99	85.48	92.4	92.5	90.34	91.03	17.59%
0 dB	72.32	85.95	96.17	81.51	83.99	81.78	69.78	84.63	85.4	80.40	82.19	17.99%
-5dB	46.63	72.31	94.5	57.75	67.80	58.93	43.53	64.83	64.48	57.94	62.87	13.64%
Average	88.49	93.31	96.77	91.86	92.61	92.15	87.61	93.12	93.06	91.49	92.05	
Rel Impr	19.33%	12.72%	29.99%	17.59%	18.81%	17.26%	17.80%	20.96%	17.40%	18.25%		18.52%

5 Conclusions

In this paper, the noise robustness of discriminative training is investigated. Discriminatively trained models are tested on both English and Chinese continuous digit databases in clean and noisy conditions. Most experiments adopt MD criterion. First, silence model should only be updated when the training and testing data are matched (Both are noisy data). Second, minimum error training is effective in noisy conditions for both clean-training and multi-training, even for SNR-based training which produces higher resolution acoustic models. Third, MD with higher error resolution than MWE is more robust in low SNR scenarios. Even when testing on mismatched noise scenarios, minimum error training is also noise robust as matched noise scenarios.

In future work, we will focus on seeking noise resistant features based on minimum error training and improve performance further in noise conditions.

References

1. Gong, Y.: Speech Recognition in Noisy Environments: A Survey. Speech Communication, Vol.16. (1995) 261-291
2. Varga, A.P., Moore, R.K.: Hidden Markov model decomposition of speech and noise. Proc. ICASSP (1990) 845-848
3. Gales, M.J.F., Young, S.J.: Robust Continuous Speech Recognition using Parallel Model Combination. Tech.Rep., Cambridge University (1994)
4. Schluter, R.: Investigations on Discriminative Training Criteria. Ph.D.thesis, Aachen University (2000)
5. Valtchev, V., Odell, J.J., Woodland, P.C., Young, S.J.: MMIE Training of Large Vocabulary Speech Recognition Systems. Speech Communication, Vol.22. 303-314
6. Juang, B.-H., Chou, W., Lee, C.-H.: Minimum Classification Error Rate Methods for Speech Recogtion. IEEE Trans. on Speech and Audio Processing, Vol.5. No.3.(1997) 257-265
7. Povey, D.: Discriminative Training for Large Vocabulary Speech Recognition. Ph.D. Thesis, Cambridge University (2004)
8. Ohkura, K., Rainton, D., Sugiyama, M.: Noise-robust HMMs Based on Minimum Error Classification. Proc.ICASSP (1993) 75-78
9. Meyer, C., Rose, G.: Improved Noise Robustness by Corrective and Rival Training. Proc.ICASSP (2001) 293-296
10. Laurila, K., Vasilache, M., Viikki, O.: A Combination of Discriminative and Maximum Likelihood Techniques for Noise Robust Speech Recognition. Proc.ICASSP (1998) 85-88
11. Kullback, S., Leibler, R.A.: On Information and Sufficiency. Ann. Math. Stat, Vol. 22. (1951) 79-86
12. Du, J., Liu, P., Soong, F.K., Zhou, J.-L., Wang, R.H.: Minimum Divergence Based Discriminative Training. Accepted by Proc.ICSLP (2006)
13. Hirsch, H.G., Pearce, D.: The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems under Noisy Conditions. In ISCA ITRW ASR2000, Paris France (2000)
14. Liu, P., Soong, F.K., Zhou, J.-L.: Effective Estimation of Kullback-Leibler Divergence between Speech Models. Tech.Rep., Microsoft Research Asia (2005)
15. Goldberger, J.: An Efficient Image Similarity Measure based on Approximations of KL-Divergence between Two Gaussian Mixtures. Proc. International Conference on Computer Vision 2003, Nice France (2003) 370-377

Distributed Speech Recognition of Mandarin Digits String

Yih-Ru Wang[1], Bo-Xuan Lu[1], Yuan-Fu Liao[2], and Sin-Horng Chen[1]

[1] National Chiao Tung Univeristy,
1001 Ta Hseuh Road, Hsinchu, 300
[2] National Taipei University of Technology,
No.1, Sec. 3, Chunghsiao E. Rd. Taipei, 106
yrwang@cc.nctu.edu.tw

Abstract. In this paper, the performance of the pitch detection algorithm in ETSI ES-202-212 XAFE standard is evaluated on a Mandarin digit string recognition task. Experimental results showed that the performance of the pitch detection algorithm degraded seriously when the SNR of speech signal was lower than 10dB. This makes the recognizer using pitch information perform inferior to the original recognizer without using pitch information in low SNR environments. A modification of the pitch detection algorithm is therefore proposed to improve the performance of pitch detection in low SNR environments. The recognition performance can be improved for most SNR levels by integrating the recognizers with and without using pitch information. Overall recognition rates of 82.1% and 86.8% were achieved for clean and multi-condition training cases.

Keywords: distributed speech recognition, Extended Advanced Front-end, tonal language speech recognition.

1 Introduction

The 3GPP had approved the Distributed Speech Recognition Extended Advanced Front-end (DSR XAFE) developed by the European Telecommunications Standards Institute STQ-Aurora working group as the recommended codec for speech enabled services on mobile appliances. In ETSI ES-202-212 DSR standard [1], the extended advanced front-end of DSR was proposed, in which (1) the algorithm for extraction of additional parameters, viz., fundamental frequency F0 (pitch) and voicing class; (2) the algorithm for pitch tracking and smoothing at the back-end to minimize pitch errors [1]. The pitch information extracted from the XAFE DSR front-end allows not only for speech reconstruction but also improved the recognition performance of tonal languages.

In the report of Aurora Group Meeting, April 2003 [2], some preliminary tonal language recognition evaluation results were given and shown that the pitch information can certainly improve the digit-string recognition performance of tonal languages such as Mandarin and Cantonese. In that report, only a clean Mandarin digit string speech database was used to evaluate the performance of tonal language

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 370 – 379, 2006.

recognition. But, the performance of the tonal language speech recognizer under noisy environment is an important issue in the real life. The environment noises will not only introduce the pitch detection errors but, more seriously, also introduce the unvoiced/voiced classification errors to significantly degrade the performance of tonal language recognition. Although many advanced pitch detectors [3] were proposed recent years which can get better performance under low SNR environment. But due to the limited computation resource in DSR front end, a modification of the XAFE front end was proposed in order to improve the performance of tonal speech recognition in this paper.

In this paper, the performance of XAFE is evaluated first in a Mandarin digit string recognition task to compare the results of using and without using the pitch information. Then, the quality of the syllable pitch contours found by XAFE is examined. A modification of XAFE is then proposed in order to improve the performance of the pitch detection algorithm in XAFE without seriously change the structure of XAFE. Lastly, the integration of both recognizers with and without using pitch information is done in order to improve the overall recognition performance.

2 Mandarin Digit String Recognition Under DSR

For evaluating the performance of DSR experimental frameworks under real environments, the Evaluations and Language resources Distribution Agency (ELDA) released a series of noisy speech databases such as: Aurora 2, Aurora SpeechDat-Car, and Aurora 4 [4]. And in [5], a Mandarin digit test set which was recorded by using an embedded PDA was used. Currently, there are still no noisy speech databases of Mandarin or other tonal languages available for evaluating the DSR performance for tonal languages. In this paper, we use a microphone-recorded Mandarin digit-string speech database to simulate the noisy speech database by adding the environment noises provided in Aurora 2.

A Mandarin digit string speech database uttered by 50 male and 50 female speakers was used in the following experiments. Each speaker pronounced 10 utterances of 1~9 digits. The speech of 90 speakers (45 male and 45 female) was used as the training set, and the other 10 speakers' as the test set. The total numbers of training and test data were 5796 and 642 syllables. The database was recorded in 16 KHz sampling rate, and then down-sampled into 8 KHz.

First, the speech spectrum features were extracted and encoded/decoded by using DSR AFE front/back-end. A 38-dimensional spectrum feature vector used in the recognizer was extracted for each 30-ms frame with 10-ms frame shift. These features were 12 MFCC, 12 delta-MFCC, 12 delta-delta-MFCC, delta-log-energy and delta-delta-log-energy. The cepstral mean normalization (CMN) technique was also used to remove the speaker effect. The Mandarin digit string recognizer then trained an 8-state HMM model for each digit. Two more models containing 3 and 1 states were also built for silence and short pause, respectively. The number of mixtures used in each state was set to 8 in this study. The performance of the recognizer is shown in Table 1(a). The average recognition rate for each noise environment shown in the table was calculated only over 5 types of SNR from 0 to 20dB. The recognition results shown in Table 1(a) are worse than the counterparts of English digit string recognition for all lower SNR cases. This maybe resulted from the existence of two confusion pairs, (one-/yi1/, seven-/qi1/) and (six-/liu4/, nine-/jiu3/) in Mandarin digit string recognition. In each pair, the two digits differ only by their consonant initials which are more easily confusing in low-SNR environment.

Then, the pitch information extracted from the XAFE was added to the recognizer aiming at improving its performance. In the DSR back-end, the pitch information extracted from XAFE front-end was smoothed using the pitch tracking and smoothing algorithm proposed in ETSI DSR standard. The pitch contour was first converted to the log-F0 contour. Then, the log pitch frequencies (log-F0) of unvoiced frames were interpolated by using exponential growth/decay functions of two nearest voiced frames [6], i.e., the log-F0 value in the n^{th} unvoiced frame can be expressed as

$$\log\left(f_0[n]\right) = MAX\left(\log\left(f_0[b]\right)\cdot e^{-\alpha(n-b)}, \log\left(f_0[e]\right)\cdot e^{-\alpha(e-n)}\right), \qquad (1)$$

where b is the frame index of the last voiced frame and e is the frame index of the next voiced frame. The attenuation factor α was set to 0.95 in this study.

Table 1(a). The recognition rate of Mandarin digit string using spectral features only

SNR (dB)	Test A				
	Subway	Babble	Car	Exhibition	Average
Clean	98.1				
20	94.9	93.3	97.0	94.7	95.0
15	90.3	91.7	95.6	91.4	92.3
10	84.4	87.5	93.8	84.7	87.6
5	66.7	77.4	86.0	70.6	75.2
0	41.1	52.0	60.4	42.1	48.9
-5	16.4	20.1	19.9	15.4	18.0
Average	75.5	80.4	86.6	76.7	79.8
SNR (dB)	Test B				
	Restaurant	Street	Airport	Train	Average
Clean	98.1				
20	90.0	95.5	90.5	95.3	92.8
15	86.1	94.4	89.4	94.9	91.2
10	80.7	87.4	86.6	90.3	86.3
5	67.0	80.2	81.5	86.3	78.8
0	48.6	48.3	57.2	65.4	54.9
-5	21.7	24.0	31.3	38.6	28.9
Average	74.5	81.2	81.0	86.4	80.8
Average for 8 types of noises with SNR in 0-20 dB					**80.3**

We then added log-F0, delta log-F0 and delta-delta log-F0 as additional recognition features. The dimension of feature vector increased to 41.

Then, a modified Mandarin digit string recognizer using pitch information was built. The number of mixtures in each state was increased to 16. The performance of the recognizer is shown in Table 1(b). In clean condition, the performance of recognizer with using pitch information was improved, because the perplexity of

Mandarin digits decreased from 10 to 2.78 when the tones of syllables were given. But, the overall recognition rate for all conditions (8 types of environment noise and 5 kinds of SNRs) was declined from 80.3% to 78.5%. This was owing to the serious performance degradations in low SNRs when the pitch information was used. Specifically, by comparing the recognition results shown in Tables 1(a) and (b), we find that the recognition rates of the recognizer using pitch information degraded significantly when the SNR of speech signal was lower than 5dB. This was especially true for the three environments of car, airport and station.

Table 1(b). The recognition rate of Mandarin digit string using both spectral and pitch features

SNR	Test A				
(dB)	Subway	Babble	Car	Exhibition	Average
Clean	98.4				
20	95.3	97.0	97.4	96.3	96.5
15	92.5	94.6	96.0	93.5	94.2
10	86.1	87.5	88.9	88.0	87.6
5	67.8	73.5	73.1	72.6	71.8
0	34.9	44.6	40.3	43.0	41.5
-5	14.5	14.2	16.7	13.2	14.7
Average	75.9	79.4	79.1	78.7	78.3
SNR	Test B				
(dB)	Restaurant	Street	Airport	Train	Average
Clean	98.4				
20	94.7	96.6	95.3	96.4	95.8
15	90.2	94.4	92.4	94.7	92.9
10	83.3	86.6	86.8	88.8	86.4
5	68.5	77.4	73.4	75.7	73.8
0	39.9	43.8	44.7	50.2	44.7
-5	17.9	19.0	17.8	29.6	21.1
Average	75.3	79.8	78.5	81.2	78.7
Average for 8 kinds of noises and 5 kinds of SNRs					**78.5**

3 Modification of the XAFE Pitch Extraction Front-End

To find the reason of performance degradation caused by adding the pitch information in low SNR speech recognition, we examined the pitch contours detected by the pitch detection algorithm of ETSI standard. Fig. 1 shows the detected pitch contours of two versions, clean and 0-dB SNR in car noise environment, of the same utterance. It can be found from the figure that lots of voiced speech frames were classified as unvoiced frames for the case of 0-dB SNR in car noise environment. Although the exponential growth/decay interpolation of pitch contour in unvoiced frames could compensate some U/V classification errors, its efficiency degraded seriously when too many voiced-to-unvoiced errors occurred. This led to the serious performance degradation for the digit string recognizer using the improper pitch information with detection errors.

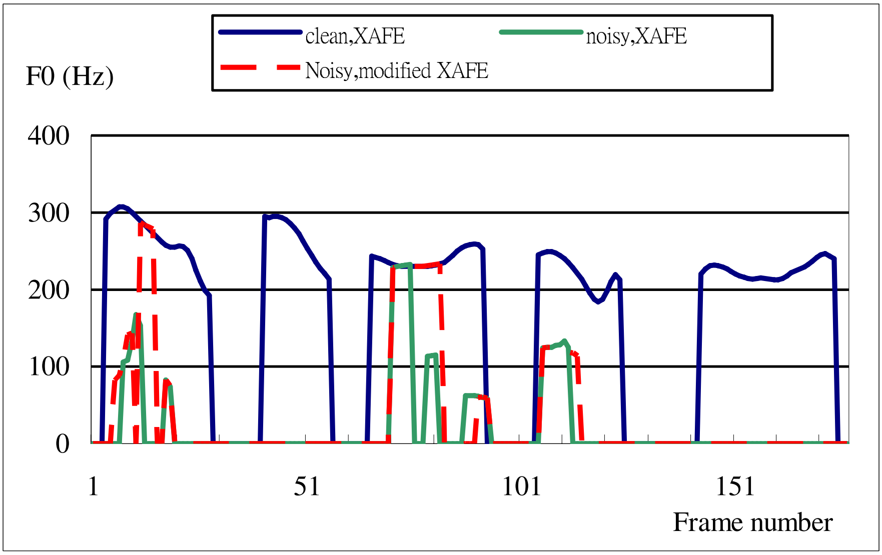


Fig. 1. An example shown the pitch contours extracted by XAFE, clean and noisy, and the pitch contours extracted by the modified pitch extraction algorithm. The speech signal condition was car environment and SNR=0dB.

By examining the ETSI pitch detection algorithm, we found that it is applied directly to the input speech. This may be the reason why it performs badly in low SNR environments. We therefore proposed to make a modification to the ETSI pitch detection algorithm by using the noise-reduced speech signal, which is of the output of the two-stage Wiener filters in the ETSI XAFE front-end, as its input. Figure 2 shows the block diagram of the modified pitch detection algorithm. Since the Wiener filters will attenuate the power of the input speech signal, a gain compensation unit was used to rise the power of the Wiener-filtered speech signal to the level of the input signal. The pitch contour of the same 0dB utterance detected by the modified pitch detection algorithm is also shown in Figure 1. It can be found from the figure that the modified pitch detection algorithm performed better for the first and third pitch segments, but the second and fifth pitch segments were still missing. So the improvement is still not significant enough to cure all V/U classification errors.

Some error analyses were done to evaluate the performances of the original and modified pitch detection algorithms operating on different noise environment with different SNR levels. Here, we took the pitch contours of the clean utterances detected by the original algorithm as the correct answers to compare. First the voiced/unvoiced classification errors were checked. Both the errors of detecting an unvoiced frame as voiced (U→V) and those of detecting a voiced frame as unvoiced (V→U) are shown in Table 2. It can be found from the table that the V→U error rate increased seriously as the SNR level decreased down below 10 dB. We also find that the modified algorithm performed slightly better for the cases with SNR level below 15 dB. The

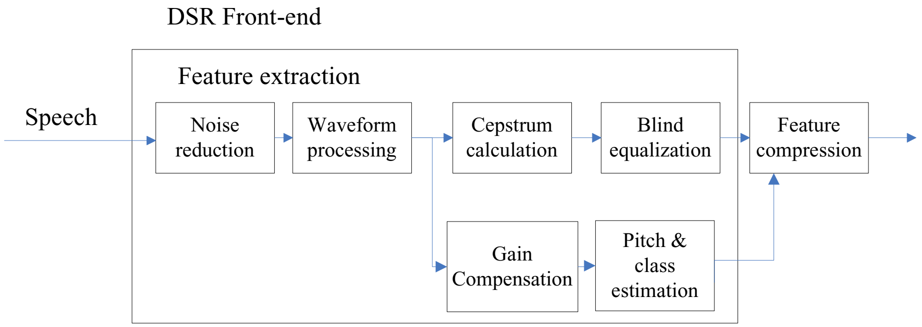


Fig. 2. The modified pitch extraction algorithm in the XAFE front end

Table 2. The comparism of XAFE pitch detection algorithm and the modified pitch detection algorithm

SNR	ETSI XAFE			Modified XAFE		
	U→V (%)	V→U (%)	V→V Pitch error	U→V (%)	V→U (%)	V→V Pitch error
20	3.64	8.71	0.008	2.68	8.74	0.010
15	3.95	13.19	0.016	2.78	13.20	0.014
10	4.19	22.03	0.037	3.15	21.41	0.033
5	4.72	41.73	0.084	3.98	39.05	0.073
0	4.51	69.65	0.136	4.21	63.96	0.113
-5	4.94	88.82	0.219	5.72	84.18	0.176

relative pitch errors, $\left|F0_{\text{clean}} - F0_{\text{noisy}}\right| / F0_{\text{clean}}$, calculated over all frames in which pitch was detected by both clean and noisy speeches were also shown in Table 2.

3.1 Performance of Mandarin Digit String Recognition Using the Modified Pitch Detection Algorithm

We then examined the performance of Mandarin digit string recognition using the pitch information extracted by the modified pitch detection algorithm. The recognition results were shown in Table 3. We find from the table that the overall recognition rate is 79.9%. By comparing the results shown in Tables 1 and 3, we find that the performance of the case using the modified pitch detection algorithm was slightly better than that of the case using the ETSI pitch detection algorithm, and was slightly worse than the case without using pitch information. By more closely examining the recognition rates in various noise environments, we find that the recognizer using the modified pitch detection algorithm performed better than the recognizer without using pitch information in high SNR, and worse in low SNR. This observation conducted to a proposal of combining the two recognizers.

Table 3. The recognition rate of Mandarin digit string using modified pitch detection algorithm

SNR	Test A				
(dB)	Subway	Babble	Car	Exhibition	Average
Clean	98.4				
20	95.3	97.7	97.7	96.0	96.7
15	93.3	93.9	96.4	93.6	94.3
10	86.9	89.1	92.2	88.8	89.3
5	71.0	75.2	77.7	75.4	74.8
0	44.1	47.8	51.9	47.7	47.9
-5	15.0	18.2	17.8	15.3	16.6
Average	78.1	80.7	83.2	80.3	80.6
SNR	Test B				
(dB)	Restaurant	Street	Airport	Train	Average
Clean	98.4				
20	92.4	96.6	93.8	96.3	94.8
15	90.3	95.5	92.7	94.2	93.2
10	80.2	87.9	88.3	86.0	85.6
5	66.4	78.2	76.6	73.8	73.8
0	43.2	46.4	50.9	54.8	48.8
-5	20.3	20.9	18.9	33.6	23.4
Average	74.5	80.9	80.5	81.0	79.2
Average for 8 kinds of noises and 5 kinds of SNRs					**79.9**

3.2 Integration of Mandarin Digit String Recognizers With and Without Using Pitch Information

As discussed above, the two recognizers with and without using pitch information were complemented to each other in higher and lower SNR environments, respectively. We therefore tried to integrate them with the goal of improving the performance for all SNR. As shown in Figure 3, the log-likelihood scores of the two recognizers were weighted combined by

$$S' = \omega \cdot S_{with\_pitch} + (1-\omega) \cdot S_{without\_pitch} \, . \tag{2}$$

Here, we let the weighting factor ω depend on the SNR d and be expressed by

$$\omega(d) = \frac{1}{1+\exp(-\gamma d + \theta)} \quad , \quad \gamma=2.5 \text{ 、 } \theta=19 \, . \tag{3}$$

An estimate of SNR can be found in the pitch detection algorithm of ETSI XAFE (Eq. (5.113) in [1]), thus we need the XAFE front-end to send the SNR information to the back end.

Table 4 shows the recognition results of the integrated recognition scheme. An overall recognition rate of 82.1% was achieved. As comparing with the recognizer without using pitch information, 1.8% recognition rate improvement was achieved. In most conditions its recognition rate tended to that of the better of the two constituent recognizers. In some cases, it performed even better than both recognizers.

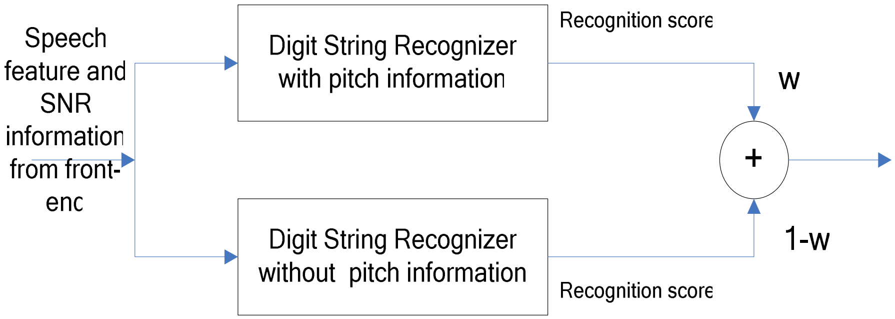

Speech feature and SNR information from front-enc
Digit String Recognizer with pitch information
Digit String Recognizer without pitch information
Recognition score
Recognition score
w
1-w
+

Fig. 3. Integration the recognizers with/without pitch information

Table 4. The recognition rate of Mandarin digit string using integration scheme

SNR (dB)	Test A				
	Subway	Babble	Car	Exhibition	Average
Clean	98.4				
20	96.0	97.5	97.8	96.3	96.9
15	93.2	94.7	96.4	93.6	94.5
10	87.7	90.2	93.8	89.9	90.4
5	71.3	78.7	84.4	73.8	77.1
0	43.6	52.8	60.1	45.2	50.4
-5	15.1	20.4	19.0	15.0	17.4
Average	78.4	82.8	86.5	79.8	81.9
SNR (dB)	Test B				
	Restaurant	Street	Airport	Train	Average
Clean	98.4				
20	92.5	96.9	93.5	96.6	94.9
15	91.6	95.5	92.7	95.0	93.7
10	83.0	88.0	89.6	88.5	87.3
5	69.0	83.0	81.5	83.5	79.3
0	50.8	48.9	57.0	65.7	55.6
-5	22.1	23.8	29.6	37.5	28.3
Average	77.4	82.5	82.9	85.9	82.2
Average for 8 kinds of noises and 5 kinds of SNRs					**82.1**

A summary of the recognition rates of all above-discussed recognition schemes under different SNRs is given in Table 5. It can be seen from the table that the performance of the proposed integrated recognition scheme was not too far away from the upper bound, 83.8%, achieved by using the pitch contours of clean speech.

Finally, the recognition results for multi-condition training condition were also calculated and shown in Table 6. Due to the size of training data, each utterance appeared twice in the training data with different noise environments or conditions. The overall recognition rate of the integrated system was 86.8%. As comparing with the recognizer without using pitch information, 10% recognition error reduction was achieved.

Table 5. Summary of recognition results of different recognizers

SNR(dB)	20	15	10	5	0	Ave.
No-pitch	93.9	91.8	87.0	77.0	51.9	80.3
With Pitch (original XAFE)	96.2	93.6	87.0	72.8	43.1	78.5
With Pitch (modified XAFE)	95.8	93.8	87.5	74.3	48.4	79.9
Integration	95.9	94.1	88.9	78.2	53.0	82.1

Table 6. Summary of recognition results of different recognizers for the case of multi-condition training

SNR(dB)		20	15	10	5	0	Ave.
No-pitch	A	96.8	95.9	92.3	83.3	57.7	85.2
	B	95.0	94.5	91.2	85.0	64.1	86.0
With Pitch (modified XAFE)	A	97.7	96.6	93.9	83.1	55.9	85.5
	B	96.5	96.0	92.5	83.0	59.0	85.4
Integration	A	97.9	96.9	94.2	84.3	58.1	86.3
	B	96.6	96.4	93.0	85.9	64.2	87.2

4 Conclusions

In this paper, the performance of Mandarin digit string recognition using ETSI XAFE front-end was carefully examined. And, due to the serious performance degeneration of pitch detector, the recognition rates of the recognizer using pitch information will degrade significantly when the SNR of the signal was lower than 5dB. A modification of the pitch detection algorithm of the XAFE standard was proposed to improve the performance of pitch detection in low SNR environments. A recognition scheme of integrating the two recognizers with and without using pitch information was also proposed to improve the recognition performance for most SNR levels. Overall recognition rates of 82.1% and 86.8% were achieved for clean and multi-condition training cases.

Acknowledgments. This work was supported by MediaTek Inc. Taiwan and NSC of Taiwan, under the project with contract NSC 94-2218-E-009 -007, 94-2213-E-009-134.

References

1. Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Extended advanced front-end feature extraction algorithm; Compression algorithms; Back-end reconstruction algorithm, ETSI Standard ES 202 212, Nov., 2003.
2. DSR Front-end Extension for Tonal-language Recognition and Speech Reconstruction. *Aurora Group Meeting*, April 2003, by IBM & Motorola, http://portal.etsi.org/stq/DSR_Presentations/Presentation.pps.

3. Wan-yi Lin and Lin-Shan Lee : Improved Tone Recognition for Fluent Mandarin Speech Based on New Inter-Syllabic Features and Robust Pitch Extraction, IEEE 8th Automatic Speech Recognition and Understanding Workshop, St. Thomas, US Virgin Islands, USA, Dec. 2003, pp. 237-242.
4. AURORA Database, http://www.elda.org/article20.html.
5. Test and Processing plan for default codec evaluation for speech enabled services (SES), Tdoc S4-030395 , 3GPP TSG SA4 meeting #26, Paris, France, 5-9 May 2003.
6. Dau-Cheng Lyu, Min-Siong Liang, Yuang-Chin Chiang, Chun-Nan Hsu, and Ren-Yuan Lyu. : Large Vocabulary Taiwanese (Min-nan) Speech Recognition Using Tone Features and Statistical Pronunciation Modeling, *Eurospeech 2003*, Geneva, pp. 1861-1864.

Unsupervised Speaker Adaptation Using Reference Speaker Weighting

Tsz-Chung Lai and Brian Mak

Department of Computer Science and Engineering
The Hong Kong University of Science & Technology
Clear Water Bay, Hong Kong
{kimo, mak}@cse.ust.hk

Abstract. Recently, we revisited the fast adaptation method called *reference speaker weighting* (RSW), and suggested a few modifications. We then showed that the algorithmically simplest technique actually outperformed conventional adaptation techniques like MAP and MLLR for 5- or 10-second supervised adaptation on the Wall Street Journal 5K task. In this paper, we would like to further investigate the performance of RSW in unsupervised adaptation mode, which is the more natural way of doing adaptation in practice. Moreover, various analyses were carried out on the reference speakers computed by the method.

1 Introduction

In practice, most automatic speech recognition systems come with a speaker-independent (SI) acoustic model that is expected to work sufficiently well with most users in general. However, the recognition performance can be further improved for a particular user if the SI model is fine-tuned to the speaking characteristics of the user through an appropriate speaker adaptation procedure. In particular, fast unsupervised speaker adaptation method that requires only a few seconds of adaptation speech from the users without knowing its content in advance is more desirable, and in some cases (e.g. phone enquiries), is the only feasible adaptation solution. Two similar fast speaker adaptation methods were proposed at about the same time: *reference speaker weighting* (RSW) [1,2] in 1997 and *eigenvoice* (EV) [3,4] in 1998. Both methods have their root in *speaker-clustering-based* methods [5]. In both methods, a speaker model is vectorized and a new speaker-adapted (SA) model is required to be a linear combination of a set of reference vectors. In eigenvoice, an orthogonal eigenspace is derived from a set of training speakers by principal component analysis, and the eigenvectors, now called *eigenvoices*, are used as the reference vectors. On the other hand, RSW simply selects a subset of training speakers as the references.

In [6], we revisited RSW with further simplifications. We also suggested to select the reference speakers by their likelihoods on the adaptation speech. Supervised adaptation using 5- and 10-second of speech on the Wall Street Journal (WSJ0) 5K-vocabulary task showed that the algorithmically simplest RSW method actually outperformed conventional adaptation methods like the

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 380–389, 2006.

Bayesian-based *maximum a posteriori* (MAP) adaptation [7], and the transformation-based *maximum likelihood linear regression* (MLLR) adaptation [8] as well as eigenvoice and eigen-MLLR [9]. Here, we would like to further our investigation on RSW by carrying out unsupervised adaptation which is the more natural way of doing adaptation in practice, as well as performing various analyses on the reference speakers computed by the method.

This paper is organized as follows. We first review the theory of reference speaker weighting (RSW) in the next Section. Unsupervised RSW adaptation was then evaluated on the Wall Street Journal corpus WSJ0 in Section 3. The experiments are followed by various analyses in Section 4. Finally, in Section 5, we present some concluding remarks.

2 Reference Speaker Weighting (RSW)

In this section, we will review the theory of reference speaker weighting in its simplest form. It is basically the same as that in [2] except with a few modifications that we have outlined in [6].

Let's consider a speech corpus consisting of N training speakers with diverse speaking or voicing characteristics. A speaker-independent (SI) model is first estimated from the whole corpus. The SI model is a hidden Markov model (HMM), and its state probability density functions are modeled by mixtures of Gaussians. Let's further assume that there are a total of R Gaussians in the SI HMM. Then, a speaker-dependent (SD) model is created for each of the N training speakers by MLLR transformation [8] of the SI model, so that all SD models have the same topology. To perform RSW adaptation, each SD model is represented by what is called a *speaker supervector* that is composed by splicing all its R Gaussian mean vectors together.

In RSW adaptation, a subset of M reference speakers $\Omega(\mathbf{s})$ is chosen among the N training speaker with $M \leq N$ for the adaptation of a new speaker $\mathbf{s}$ as depicted in Fig. 1. (Notice that the set of reference speakers, in general, is different for each new speaker.) Let $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M\}$ be the set of reference speaker supervectors. Then the RSW estimate of the new speaker's supervector is

$$\mathbf{s} \approx \mathbf{s}^{(rsw)} = \sum_{m=1}^{M} w_m \mathbf{y}_m = \mathbf{Y}\mathbf{w} \ , \tag{1}$$

and for the mean vector of the rth Gaussian,

$$\mathbf{s}_r^{(rsw)} = \sum_{m=1}^{M} w_m \mathbf{y}_{mr} = \mathbf{Y}_r \mathbf{w} \ . \tag{2}$$

where $\mathbf{w} = [w_1, w_2, \ldots, w_M]'$ is the combination weight vector.

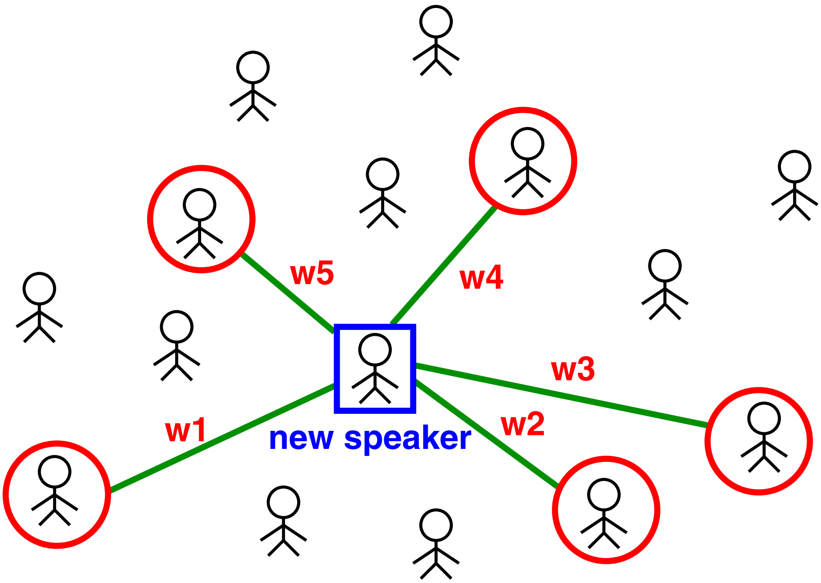


Fig. 1. Concept of reference speaker weighting

2.1 Maximum-Likelihood Estimation of Weights

Given the adaptation data $\mathbf{O} = \{\mathbf{o}_t, t = 1, \ldots, T\}$, one may estimate $\mathbf{w}$ by maximizing the following $Q(\mathbf{w})$ function:

$$Q(\mathbf{w}) = -\sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_t(r)(\mathbf{o}_t - \mathbf{s}_r^{(rsw)}(\mathbf{w}))' \mathbf{C}_r^{-1} (\mathbf{o}_t - \mathbf{s}_r^{(rsw)}(\mathbf{w}))$$

where $\gamma_t(r)$ is the posterior probability of observing $\mathbf{o}_t$ in the rth Gaussian, and $\mathbf{C}_r$ is the covariance matrix of the rth Gaussian. The optimal weight vector may be found by simple calculus as follows:

$$\frac{\partial Q}{\partial \mathbf{w}} = 2 \sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_t(r) \mathbf{Y}_r' \mathbf{C}_r^{-1} (\mathbf{o}_t - \mathbf{Y}_r \mathbf{w}) = 0$$

$$\Rightarrow \mathbf{w} = \left[\sum_{r=1}^{R} \left(\sum_{t=1}^{T} \gamma_t(r) \right) \mathbf{Y}_r' \mathbf{C}_r^{-1} \mathbf{Y}_r \right]^{-1} \left[\sum_{r=1}^{R} \mathbf{Y}_r' \mathbf{C}_r^{-1} \left(\sum_{t=1}^{T} \gamma_t(r) \mathbf{o}_t \right) \right] . \quad (3)$$

Thus, the weights $\mathbf{w}$ may be obtained by solving a system of M linear equations. The solution requires finding the inverse of an $M \times M$ matrix and has a computational complexity of $O(M^3)$. Notice also that unlike Hazen's formulation in [2], no constraints are imposed on the combination weights.

2.2 Maximum-Likelihood Reference Speakers

In [6], we showed that good RSW adaptation performance could be achieved by selecting those training speakers that gave the highest likelihoods of the adaptation speech from a test speaker as his/her reference speakers. We call these reference speakers the maximum-likelihood (ML) reference speakers. We continue to use ML reference speakers for RSW adaptation evaluation in this paper.

3 Experimental Evaluation

Unsupervised fast speaker adaptation was carried out on the Wall Street Journal WSJ0 [10] 5K-vocabulary task using our modified reference speaker weighting (RSW) method.

Table 1. Duration statistics (in seconds) of the test utterances of each WSJ0 test speaker

Speaker ID	#Utterances	min	max	mean	std dev
440	40	4.32	12.76	8.23	6.83
441	42	2.82	10.94	6.89	4.14
442	42	2.42	11.85	7.24	4.62
443	40	3.34	14.19	8.01	5.49
444	41	2.36	11.13	7.88	4.49
445	42	2.55	10.70	5.81	4.18
446	40	2.99	12.28	7.14	5.19
447	43	2.06	11.55	7.33	5.78

3.1 WSJ0 Corpus and the Evaluation Procedure

The standard SI-84 training set was used for training the speaker-independent (SI) model and gender-dependent (GD) models. It consists of 83 speakers (41 male speakers and 42 female speakers) and 7138 utterances for a total of about 14 hours of training speech. The standard nov'92 5K non-verbalized test set was used for evaluation. It consists of 8 speakers (5 male and 3 female speakers), each with about 40 utterances. The detailed duration statistics of the test utterances of each speaker is given in Table 1.

During unsupervised adaptation, the content of each test utterance was not assumed to be known in advance. All adaptation methods under investigation were run with 3 EM iterations. During each iteration, the current speaker-adapted (SA) model was used to decode the adaptation utterance and to provide the Gaussian mixture posterior probabilities, then the adaptation method was carried out to get a new SA model. At the first iteration, the SI model was used for decoding instead. The last SA model was used to decode the same utterance again to produce the final recognition output. Results from all speakers and all utterances are pooled together and their average results are reported. Finally, a bigram language model of perplexity 147 was employed in this recognition task.

3.2 Acoustic Modeling

The traditional 39-dimensional MFCC vectors were extracted at every 10ms over a window of 25ms. The speaker-independent (SI) model consists of 15,449 cross-word triphones based on 39 base phonemes. Each triphone was modeled as a continuous density HMM (CDHMM) which is strictly left-to-right and has three states with a Gaussian mixture density of 16 components per state; there are 3,131 tied states in total. The SI model has a word recognition accuracy of 92.60% on the test data[1]. GD models were then created by MAP adaptation from the SI model using gender-specific training data, and they give a word recognition accuracy of 92.92%.

Furthermore, 83 speaker-dependent (SD) models were created by MLLR adaptation using a regression class tree of 32 classes for RSW adaptation methods.

3.3 Effect of the Number of Reference Speakers

We first investigate how many ML reference speakers are sufficient for RSW adaptation. Unsupervised RSW adaptation using only a single utterance at a time was performed with 3 EM iterations. We started with 10 ML reference speakers and then doubled the number until all 83 training speakers were used. The results are plotted in Fig. 2. The figure shows that although using **all** training speakers as reference speakers gives good results, the best adaptation performance actually is obtained with 40 reference speakers, though the difference is small. It also shows that RSW performance saturates fast after about half of training speakers are used as reference speakers.

3.4 Comparative Study

RSW adaptation was compared with the following models and common adaptation methods:

SI: the SI model.
GD: the gender-dependent models.
MAP: the SA model found by MAP adaptation [7].
MLLR: the SA model found by MLLR adaptation [8].

For the evaluation using GD models, the test speaker's gender was assumed known and the GD model of the corresponding gender was applied to his/her utterances; thus, there is no error from gender detection[2]. For each adaptation method, we tried our best effort to get the best performance. MAP and MLLR were performed using HTK. For MAP, scaling factors in the range of 3–15 were attempted, but none of them gave any improvement; MLLR made use of a regression tree of 32 regression classes (though it was found actually in most cases, only a single global transform was employed) and block-diagonal transforms (with 3 blocks)

[1] The accuracy of the SI model is better than what we had reported in [6] because better values of grammar factor and insertion penalty are used.

[2] As will be explained in Section 4.2, the gender of speaker 442 is actually female.

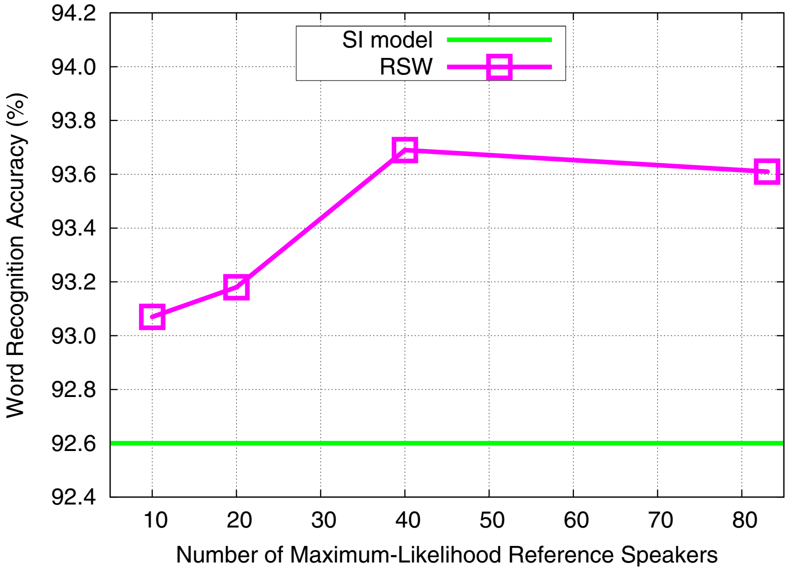


Fig. 2. Effect of the number of ML reference speakers on RSW

Table 2. Comparing RSW with the SI and GD models, MAP and MLLR adaptation on WSJ0. Results are word accuracies in %. (WERR is word error rate reduction in %, and M is the number of reference speakers.)

Model/Method	Word Accuracy	WERR
SI	92.60	—
GD	92.98	5.14
MAP	92.60	0.0
MLLR (3 blocks)	93.24	8.65
RSW (M=10)	93.07	6.35
RSW (M=20)	93.18	7.84
RSW (M=40)	**93.69**	**14.7**
RSW (M=83)	93.61	13.6

as there were no improvement from using full-MLLR transforms; finally, RSW adaptation using 10, 20, 40, and 83 ML reference speakers was attempted for the comparison. Again, each time, only a single utterance was used for unsupervised adaptation and 3 EM iterations were run. The results are summarized in Table 2.

From Table 2, we are again surprised that the algorithmically simplest RSW technique actually gives the best fast adaptation performance.

3.5 Saturation Effect of RSW

A more detailed look at the adaptation performance of MLLR and RSW (using 40 ML reference speakers) across the three EM iterations is shown in Fig. 3.

It can be seen that MLLR does not improve much after the first iteration and RSW saturates after the second iteration.

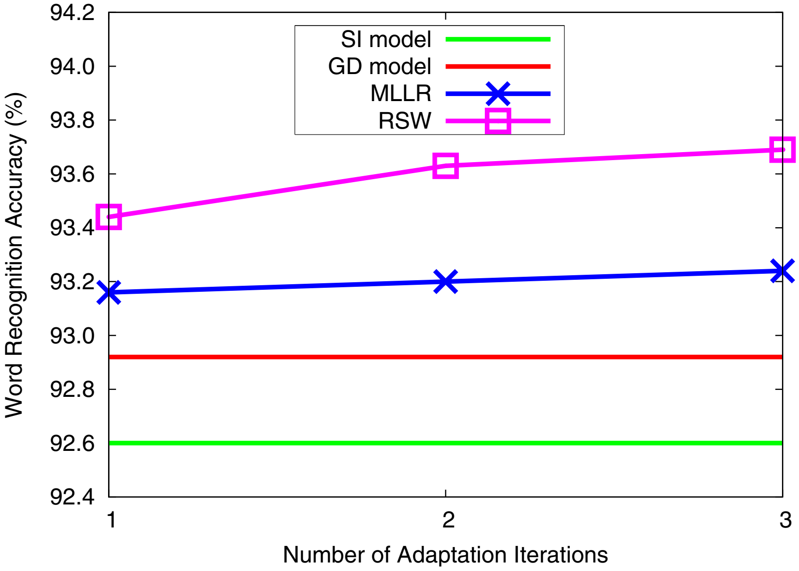


Fig. 3. Saturation effect of MLLR and RSW adaptation on WSJ0

4 Analysis

In this section, we would like to analyze the maximum-likelihood (ML) reference speakers found for each test speaker from each test utterance.

4.1 Consistency of ML Reference Speakers

Since each test speaker has about 40 test utterances for adaptation, it will be interesting to see how likely that the same ML reference speakers are selected for each test utterance of the same test speaker. To do that, during unsupervised RSW adaptation using M reference speakers, the M ML reference speakers of each test utterance were recorded. Then all the ML reference speakers over all test utterances of the same test speaker are sorted according to their frequencies. Finally the total frequency of the M most frequent reference speakers are found and the percentage of their contribution over all the reference speakers is computed. The percentage is used as a measure of how consistent are the ML reference speakers found by using any utterance of a test speaker. The reference speaker consistency percentages for each test speaker is summarized in Table 3.

From Table 3, we can see that the consistency is quite high. We may conclude that (1) finding reference speakers by maximizing the likelihood of a test speaker's adaptation speech is effective, and (2) one may find the ML reference speakers using *any* utterance of a test speaker.

Table 3. Consistency percentage of ML reference speakers found for each WSJ0 test speaker. (M is the number of reference speakers.)

Speaker ID	M = 10	M = 20	M = 40
440	0.830	0.854	0.908
441	0.790	0.820	0.903
442	0.860	0.863	0.903
443	0.853	0.848	0.978
444	0.868	0.870	0.915
445	0.919	0.920	0.899
446	0.815	0.794	0.887
447	0.865	0.887	0.916
Overall	0.850	0.857	0.913

4.2 Consistency of ML Reference Speakers' Gender

Since gender is generally considered as a major factor affecting one's voicing characteristics, it is interesting to see if one's reference speakers have the same gender as oneself. Here is our analysis procedure: from the RSW unsupervised adaptation using M reference speakers of each of the N test utterances of a test speaker, there are totally MN reference speakers; among those MN reference speakers, count how many of them have the same gender as the test speaker's, and compute their ratio which we call the *gender consistency percentage.* Table 4 lists out the gender consistency percentages of all the 8 test speakers.

We find that when 10 reference speakers are employed by RSW adaptation, half of the 8 test speakers have a gender consistency percentage close to 100%. The consistency percentage is particular bad for the test speaker labeled as 442. However, after we listened to speaker 442's utterances, we believe that there is an error in the gender label and the speaker is actually a female. The last row of Table 4 is obtained by correcting speaker 442's gender to female. On the other

Table 4. Consistency pencentage of the gender of ML reference speakers found for each WSJ0 test speaker. (M is the number of reference speakers.)

Speaker ID	Gender	M = 10	M = 20	M = 40
440	male	0.958	0.920	0.773
441	female	0.993	0.956	0.830
442	male	0.260	0.365	0.368
443	male	1.000	1.000	0.899
444	female	0.998	0.968	0.812
445	female	0.902	0.893	0.758
446	male	0.803	0.708	0.613
447	male	0.991	0.986	0.833
Overall	—	0.862	0.849	0.735
442 as female	—	0.923	0.883	0.769

hand, as expected, the gender consistency percentage drops as more reference speakers are employed in RSW adaptation. The high percentages suggest that (1) the common use of gender-dependent models for speech recognition is sensible, and (2) our approach of finding ML reference speakers may be modified to a gender detection method.

5 Conclusions

In this paper, we show that reference speaker weighting is effective for fast speaker adaptation in unsupervised mode as well as in supervised mode (the latter had been investigated in [6]). Its performance is better than MAP and MLLR on WSJ0 when only one utterance is available for unsupervised adaptation. It is also a very simple algorithm. It is also found that it is not necessary to use all training speakers as reference speakers and for this particular task, using half of training speakers actually gives slightly better adaptation results. Analyses on the reference speakers found using ML criterion show that the chosen reference speakers are very consistent across utterances from the same speaker in terms of their identity or gender.

Acknowledgments

This research is partially supported by the Research Grants Council of the Hong Kong SAR under the grant number DAG05/06.EG43.

References

1. Tim J. Hazen and James R. Glass, "A comparison of novel techniques for instantaneous speaker adaptation," in *Proceedings of the European Conference on Speech Communication and Technology*, 1997, pp. 2047–2050.
2. Tim J. Hazen, "A comparison of novel techniques for rapid speaker adaptation," *Speech Communications*, vol. 31, pp. 15–33, May 2000.
3. R. Kuhn, P. Nguyen, J.-C. Junqua, et al., "Eigenvoices for speaker adaptation," in *Proceedings of the International Conference on Spoken Language Processing*, 1998, vol. 5, pp. 1771–1774.
4. H. Botterweck, "Very fast adaptation for large vocabulary continuous speech recognition using eigenvoices," in *Proceedings of the International Conference on Spoken Language Processing*, 2000, vol. 4, pp. 354–357.
5. T. Kosaka, S. Matsunaga, and S. Sagayama, "Speaker-independent speech recognition based on tree-structured speaker clustering," *Journal of Computer Speech and Language*, vol. 10, pp. 55–74, 1996.
6. Brian Mak, Tsz-Chung Lai, and Roger Hsiao, "Improving reference speaker weighting adaptation by the use of maximum-likelihood reference speakers," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Toulouse, France, May 14–19 2006.
7. J. L. Gauvain and C. H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, April 1994.

8. C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Journal of Computer Speech and Language*, vol. 9, pp. 171–185, 1995.
9. K. T. Chen, W. W. Liau, H. M. Wang, and L. S. Lee, "Fast speaker adaptation using eigenspace-based maximum likelihood linear regression," in *Proceedings of the International Conference on Spoken Language Processing*, 2000, vol. 3, pp. 742–745.
10. D. B. Paul and J. M. Baker, "The design of the Wall Street Journal-based CSR corpus," in *Proceedings of the DARPA Speech and Natural Language Workshop*, Feb. 1992.

Automatic Construction of Regression Class Tree for MLLR Via Model-Based Hierarchical Clustering

Shih-Sian Cheng [1,2], Yeong-Yuh Xu[1], Hsin-Min Wang[2], and Hsin-Chia Fu[1]

[1] Department of Computer Science, National Chiao-Tung University, Hsinchu
{yyxu, hcfu}@csie.nctu.edu.tw
[2] Institute of Information Science, Academia Sinica , Taipei
{sscheng, whm}@iis.sinica.edu.tw

Abstract. In this paper, we propose a model-based hierarchical clustering algorithm that automatically builds a regression class tree for the well-known speaker adaptation technique - Maximum Likelihood Linear Regression (MLLR). When building a regression class tree, the mean vectors of the Gaussian components of the model set of a speaker independent CDHMM-based speech recognition system are collected as the input data for clustering. The proposed algorithm comprises two stages. First, the input data (i.e., all the Gaussian mean vectors of the CDHMMs) is iteratively partitioned by a divisive hierarchical clustering strategy, and the Bayesian Information Criterion (BIC) is applied to determine the number of clusters (i.e., the base classes of the regression class tree). Then, the regression class tree is built by iteratively merging these base clusters using an agglomerative hierarchical clustering strategy, which also uses BIC as the merging criterion. We evaluated the proposed regression class tree construction algorithm on a Mandarin Chinese continuous speech recognition task. Compared to the regression class tree implementation in HTK, the proposed algorithm is more effective in building the regression class tree and can determine the number of regression classes automatically.

Keywords: speaker adaptation, MLLR, regression class tree.

1 Introduction

MLLR [1] is well known for its ability to perform rapid and robust speaker adaptation with a small amount of adaptation data. Extensive research efforts have been made to improve MLLR [8, 13] as well as to develop new methods that extend the conventional MLLR framework [2-7].

In the MLLR proposed by Leggetter and Woodland [1], adaptation of speaker independent (SI) model parameters (e.g., the mean parameters of a CDHMM-based speech recognition system) is carried out via a set of linear transformations, where each regression (transformation) matrix is responsible for the adaptation of one regression class (subset of the model parameters). To enhance flexibility and robustness, the authors proposed using of a regression class tree to group the parameters of the model set into regression classes. The purpose is to dynamically determine the sharing of regression matrices for the parameters according to the

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 390 – 398, 2006.

amount and type of adaptation data available [8]. The regression class tree is a critical component in the MLLR framework as well as in other linear transformation based approaches, e.g., [3].

The issue of regression class tree construction for MLLR can be viewed as a data clustering problem of the parameters. For example, HTK [9] applies a centroid splitting algorithm to construct a regression class tree, in which the number of base clusters (classes) must be determined empirically. In this study, we developed a model-based hierarchical clustering algorithm, which not only provides a better clustering result for the model parameters, but also determines the number of clusters (i.e., base classes of the regression class tree) automatically. The proposed regression class tree construction algorithm is a two-stage process. In the first stage, the input data is iteratively partitioned in a top-down fashion using a divisive hierarchical clustering strategy, and the Bayesian Information Criterion (BIC) [10] is applied to determine the number of clusters. In the second stage, these clusters are iteratively merged in a bottom-up fashion to build the regression class tree. To evaluate the performance, the proposed regression class tree implementation was compared with that of HTK. The experimental results show that the proposed algorithm is effective in building a regression class tree automatically and in determining the number of regression classes for MLLR.

The rest of this paper is organized as follows. First, MLLR and the concept of regression class tree are reviewed in Section 2. Then, the proposed algorithm for regression class tree construction is introduced in Section 3. The experimental results are presented in Section 4, followed by our conclusions in Section 5.

2 MLLR and Regression Class Tree

In MLLR, to adapt the SI Gaussian mean vectors for example, the mean vectors are clustered into C regression classes, and each regression class c is associated with an n×(n+1) regression matrix $\mathbf{W}_c$, where n is the dimensionality of the feature vector. Let the mean vector $\boldsymbol{\mu}_m = [\mu_m(1),\ldots, \mu_m(n)]^T$ of Gaussian component m be one of the T_c mean vectors in the regression class c; then, the adapted mean vector can be derived as

$$\hat{\boldsymbol{\mu}}_m = \mathbf{W}_c \xi_m = \mathbf{A}_c \boldsymbol{\mu}_m + \mathbf{b}_c, m = 1,\ldots,T_c; c = 1,2,\ldots,C, \tag{1}$$

where $\xi_m=[1, \mu_m(1),\ldots, \mu_m(n)]^T$ is the (n+1)-dimensional augmented mean vector. $\mathbf{A}_c$ and $\mathbf{b}_c$ are an n×n matrix and an n-dimensional vector, respectively, such that $\mathbf{W}_c$ = $[\mathbf{b}_c \ \mathbf{A}_c]$. $\mathbf{b}_c$ is used as a bias vector. $\mathbf{A}_c$ can be diagonal, block-diagonal, or full. $\{\mathbf{W}_c\}_{c=1,\ldots,C}$ is estimated by maximizing the likelihood of the adaptation data for the adapted parameters using EM algorithm.

To facilitate flexibility and robustness, MLLR usually makes use of a regression class tree. All the Gaussian components are arranged into a tree, which is basically a binary tree, such that close components in the acoustic space are grouped in the same node (regression class). The lower level of the tree indicates that the components are more close. In the hierarchy of the tree, each parent node contains all the components of its two child nodes, and all the leaf nodes are termed as *base classes*. During the adaptation process, the feature vectors used for adaptation are aligned to the corresponding Gaussian components, and the occupation counts are accumulated for each of the base classes. The regression class tree can be traversed in either a top-down or a bottom-up fashion to only generate transformations for those nodes that have sufficient adaptation data. Fig. 1 shows an example of a regression class tree. The numbers in italics associated with the tree nodes are the number of adaptation feature vectors aligned to them. If the threshold for the sufficiency of the adaptation data is set as 300, only the transformations for regression nodes 2, 3, and 4 will be constructed. The transformation of node 2 will take charge of the adaptation of Gaussian components in node 5, and the transformation of noe 3 will take charge of nodes 6 and 7.

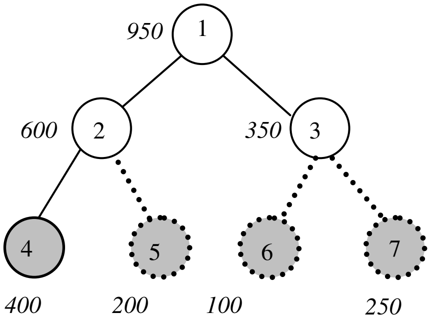


Fig. 1. An example of a regression class tree

3 Model-Based Hierarchical Clustering for Automatic Regression Class Tree Construction

In this section, before describing the proposed regression class tree construction algorithm in detail, we briefly introduce BIC, which provides the splitting and merging criteria for the proposed algorithm.

3.1 Model Selection and BIC

Given a data set $X=\{x_1, x_2,\ldots, x_n\}$ and a set of candidate models $M=\{M_1, M_2,\ldots, M_k\}$, the model selection problem is to choose the model that best fits the distribution of X. BIC is a model selection criterion and the BIC value of model M_i is

$$BIC(M_i, X) = \log p(X \mid \hat{\Theta}_i) - \frac{1}{2}\#(M_i)\log n, \tag{2}$$

where $p(X \mid \hat{\Theta}_i)$ is the maximum likelihood of X for model M_i, and $\#(M_i)$ is the number of parameters of M_i. The model with the highest BIC value is selected. The BIC-based approach is also known as a penalized likelihood approach, which gives a larger penalty to more complex models.

3.2 The Proposed Regression Class Tree Construction Algorithm

The proposed regression class tree construction algorithm is a two-stage process. In the first stage, the input data *X* is viewed as a single cluster initially, after which the clusters are divided into finer clusters iteratively by using BIC as the validity criterion for splitting until there is no cluster should be split. Then, in the second stage, similar to agglomerative hierarchical clustering, these clusters are iteratively merged in a bottom-up fashion to build the resultant dendrogram. The details of the proposed clustering algorithm are given in Algorithm 1, which we call TDBU (Top-Down & Bottom-Up). There are two major issues with respect to the proposed clustering algorithm:

(***I1***) In the Top-Down (TD) stage, which cluster should be split into a pair of sub-clusters and how should it be split?
(***I2***) In the Bottom-Up (BU) stage, what is the appropriate distance measure of two clusters and how should they be merged?

On Issue (***I1***)
At each splitting iteration, each cluster C_i with $\Delta BIC_{21}(C_i)=BIC(GMM_2, C_i) - BIC(GMM_1, C_i)$ larger than 0 is split into two sub-clusters, where GMM_k represents a Gaussian mixture model with *k* mixture components. According to BIC theory, the larger the value of $\Delta BIC_{21}(C_i)$, the better GMM_2 will fit C_i, and thus the more confidence there will be that C_i is composed of at least two Gaussian clusters. As to the splitting of cluster C_i, after the training of GMM_2, each sample belonging to C_i is distributed to the Gaussian component that has the largest posterior probability for the sample. In other words, suppose Θ_1 and Θ_2 are the two components of GMM_2, for each *x* in C_i, then *x* is distributed to $cluster_{\Theta j}$ if $j=\arg\max_r p(\Theta_r|x)$.

On issue (***I2***)
At each merging iteration in the second stage, the two most similar (close) clusters are merged into a single cluster. Given two clusters, C_i and C_j, let $C'=\{ C_i , C_j \}$. Then, $\Delta BIC_{21}(C')$ is used to represent the dissimilarity (or distance) between C_i and C_j. The smaller the $\Delta BIC_{21}(C')$ value the more confident we are in describing the distribution of C′ as one Gaussian cluster.

In the proposed TDBU algorithm, the TD stage alone can construct a regression class tree. However, the regression class tree constructed by the following BU stage is believed to be better than that constructed by the TD stage alone. The TD stage can capture the real clusters in *X* approximately, but may not construct an optimal dendrogram for the real clusters because of the uncertainties of the splitting processes and the suboptimal hierarchy construction of the clusters. We consider that the major contribution of the TD stage is to automatically determine the number of clusters in *X* and to provide a decent clustering result for the BU stage to start with. After the TD stage, the BU stage can construct a better hierarchy for these clusters, since it proceeds as the conventional (non-model-based) hierarchical agglomerative clustering. Fig. 2 illustrates the clustering process of the TDBU algorithm with a simple example. We can clearly see the differences between the dendrograms constructed by the TD stage alone and by the complete TDBU process. The memory complexity of the BU stage for storing the distance matrix is $O(m^2)$, where m is the number of clusters produced by the TD stage, compared to $O(n^2)$ for the conventional hierarchical agglomerative clustering approach, where n is the number of input samples. Obviously, $O(m^2)$ is smaller than $O(n^2)$.

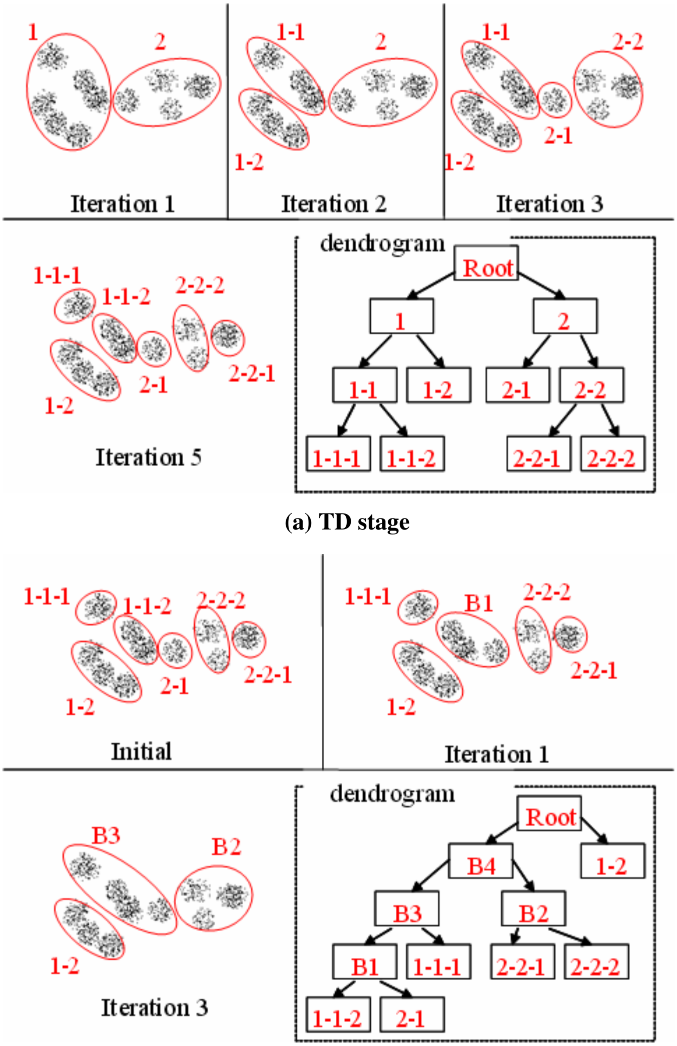
```
Algorithm: TDBU
Input: Data set $X=\{x_1, x_2, \ldots, x_n\}$.
Output: A dendrogram of the input data set $X$.
Begin
  Top-Down (TD) stage:
    1. Start with one single cluster (the root node of the TD
       dendrogram).
    2. Repeat:
         Split cluster (leaf node) $C_i$ with $\Delta BIC_{21}(C_i)>0$ into
         two new clusters (leaf nodes).
       Until there is no cluster (leaf node) whose $\Delta BIC_{21}$
       value is larger than 0.
  Bottom-Up (BU) stage:
    1. Start with the resultant clusters $C_1, C_2, \ldots, C_m$ in the
       TD stage (the leaf nodes of TD dendrogram).
    2. Repeat:
         Merge the two closest clusters (nodes) into a single
         cluster (parent node) at the next level of the BU
         dendrogram.
       Until only one cluster (root node) left.
    3. Output the BU dendrogram.
End
```

Algorithm 1. The proposed model-based hierarchical clustering algorithm for MLLR regression class tree construction

4 Experiments

4.1 Experimental Setup

The proposed approach was evaluated on the TCC300 continuous Mandarin Chinese microphone speech database [12], which contains data of 150 female and 150 male speakers. The speech data of 260 speakers, a total of 23.16 hours was used to train the SI acoustic model, while the speech data of eight speakers (four female and four male), not included in the 260 training speakers was used for model adaptation and testing. The sampling rate of the speech was 16 kHz. Twelve MFCCs and log-energy, along with their first and second order time derivatives, were combined to form a 39-dimensional feature vector. Utterance-based Cepstral mean subtraction (CMS) was applied to the training and test speech to remove the channel effect.



(a) TD stage

(b) BU stage

Fig. 2. An example of the TDBU clustering process. The resultant clusters at iteration 5 of the TD stage are fed to the BU stage as the initial condition. The dendrogram constructed in the BU stage is the output of TDBU.

Considering the monosyllabic structure of the Chinese language in which each syllable can be decomposed into an INITIAL/FINAL format, the acoustic units used in our speech recognizer are intra-syllable right-context-dependent INITIAL/FINAL, including 112 context-dependent INITIALs and 38 context-independent FINALs [11]. Each INITIAL is represented by a CDHMM with three states, while each FINAL is represented with four states. The number of Gaussian components for each state is 32. For each test speaker, about 125 seconds of speech data was used for model adaptation, while 400 seconds was used for speech recognition evaluation. In the adaptation experiments, the 125-second adaptation speech for each test speaker was averagely chopped into 25 five-second utterances. The recognizer performed only free syllable decoding without any grammar constraints. Syllable accuracy was used as the evaluation metric. All adaptation experiments were conducted in a supervised manner and only mean vectors of Gaussian components in the SI model were adapted. The speaker independent recognition accuracy was 66.20%, averaged over the eight test speakers. The performance of the built-in approach in HTK [9] was used as the baseline result. The speaker adaptation experiments on the proposed approach were also performed with HTK.

4.2 Experimental Results

Fig. 3 shows the adaptation performance of various regression class trees constructed by the built-in HTK approach and the proposed algorithm - TDBU. The number of base classes predefined for HTK ranged from 4 (denoted as HTK4) to 200 (denoted as HTK200). Full-covariance Gaussians were used to compute the ΔBIC value in the TDBU approach, and the number of base classes automatically determined by TDBU was 34. For each test speaker, the 25 five-second utterances were used for adaptation in order. For example, if the number of utterances is five, the adaptation was performed on the first five utterances.

Several conclusions can be drawn from Fig. 3: (1) When the amount of adaptation data is small (less than 10 utterances), there is no significant difference between the performance of all the approaches tested due to the very limited adaptation data. (2) If more adaptation data (more than 10 utterances) is available, the performance can be improved with more complex regression class trees (more base classes). 34 seems to be an appropriate number of base classes since the performance of HTK34, HTK64, and HTK200 is almost the same and are superior to the results obtained with fewer base classes. (3) It is clear that TDBU34 outperforms HTK34, HTK64, and HTK200. The experiment results show that the TDBU approach is not only more effective than the regression class tree implementation method in HTK, but can also find an appropriate number of base classes automatically during the regression class tree construction process. This is an advantage when we need to take account of the memory requirement of the regression class tree when designing an embedded speech recognition system for a device with limited memory.

As mentioned in Section 3, the TD stage (i.e., the first stage of TDBU) can be used alone to construct the regression class tree. Fig. 4 depicts the performance curves of TD34, TDBU34 and HTK34, from which we can infer that performing the BU stage after the TD stage definitely constructs a better hierarchy for the regression classes than that constructed using the TD stage alone. The experiment results also show that, in general, the TD34 regression class tree outperforms the HTK34 regression class tree.

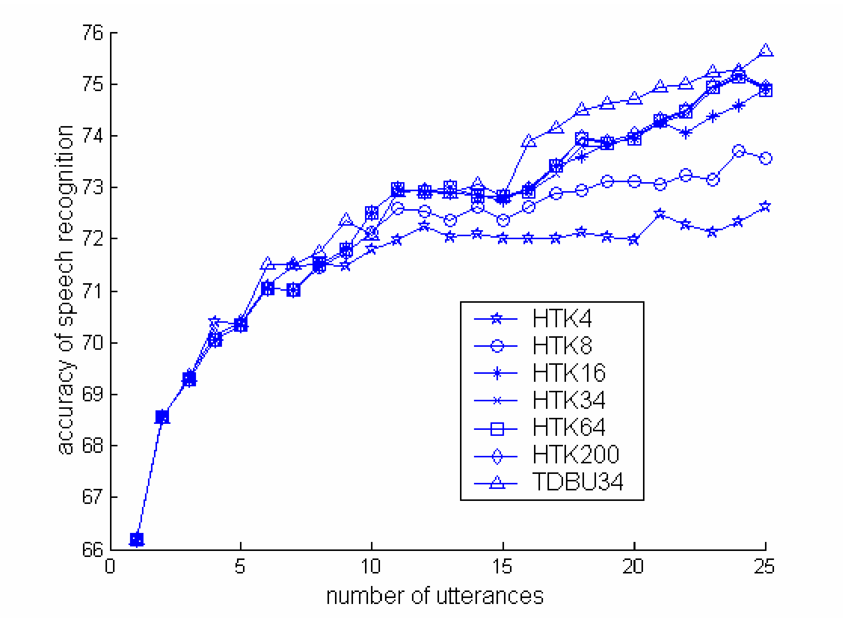


Fig. 3. Adaptation performance obtained with various regression class trees constructed by HTK and TDBU. The number of base classes determined by TDBU is 34.

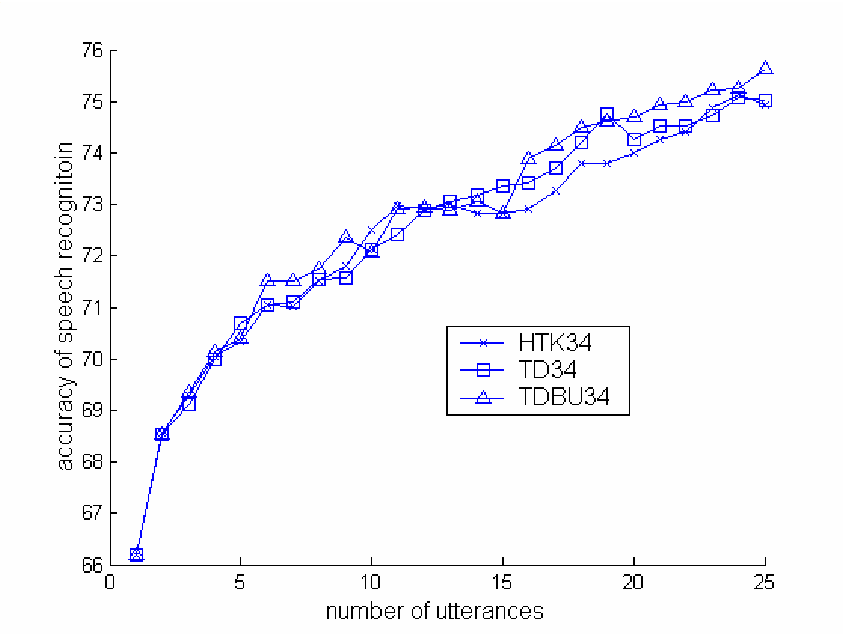


Fig. 4. Adaptation performance of HTK34, TD34 and TDBU34

5 Conclusion

This paper presents a model-based hierarchical clustering algorithm for MLLR regression class tree construction. The experiment results shows that the regression class tree constructed by our approach is more effective than that constructed by HTK. In addition, our approach can automatically decide an appropriate number of regression classes, which used to be decided empirically.

References

1. Leggetter, C. J. and Woodland, P. C.: Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models. Computer Speech and Language, vol. 9 (1995) 171-185.
2. Chesta, C., Siohan, O., and Lee, C.-H.: Maximum a Posteriori Linear Regression for Hidden Markov Model Adaptation. Proc. EUROSPEECH'1999.
3. Siohan, O., Myrvoll, T.-A., and Lee, C.-H.: Structural Maximum a Posteriori linear Regression for Fast HMM Adaptation. Workshop on Automatic Speech Recognition 2000. ISCA ITRW ASR'2000.
4. Chen, K. T., Liau, W. W., Wang, H. M., and Lee, L. S.: Fast Speaker Adaptation Using Eigenspace-based Maximum Likelihood Linear Regression. Proc. ICSLP'2000.
5. Chen, K. T. and Wang, H. M.: Eigenspace-based Maximum a Posteriori Linear Regression for Rapid Speaker Adaptation. Proc. ICASSP'2001.
6. Doumpiotis, V. and Deng, Y.: Eigenspace-based MLLR with Speaker Adaptive Training in Large Vocabulary Conversation Speech Recognition. Proc. ICASSP'2004.
7. Mak. B. and Hsiao. R.: Improving Eigenspace-based MLLR Adaptation by Kernel PCA. Proc. ICSLP'2004.
8. Leggetter, C. J. and Woodland, P.C.: Flexible Speaker Adaptation Using Maximum Likelihood Linear Regression. Proc. ARPA Spoken Language Systems Technology Workshop, 1995.
9. HTK Speech Recognition Toolkit, http://htk.eng.cam.ac.uk/
10. Fraley, C. and Raftery, A. E.: How Many Clusters? Which Clustering Method? Answers via Model-based Cluster Analysis. Computer Journal, 41 (1998) 578-588.
11. Wang, H. M. et al.: Complete Recognition of Continuous Mandarin Speech for Chinese Language with Very Large Vocabulary Using Limited Training Data. IEEE Trans. on Speech and Audio Proc., 5(2) (1997) 195-200.
12. The Association for Computational Linguistics and Chinese Language Processing, http://www.aclclp.org.tw/corp.php
13. Gales, M. J. F.: Maximum Likelihood Linear Transformations for HMM-based Speech Recognition. Computer Speech and Language, vol. 12 (1998) 75-98.

A Minimum Boundary Error Framework for Automatic Phonetic Segmentation

Jen-Wei Kuo and Hsin-Min Wang

Institute of Information Science, Academia Sinica, Taipei
{rogerkuo, whm}@iis.sinica.edu.tw

Abstract. This paper presents a novel framework for HMM-based automatic phonetic segmentation that improves the accuracy of placing phone boundaries. In the framework, both training and segmentation approaches are proposed according to the minimum boundary error (MBE) criterion, which tries to minimize the expected boundary errors over a set of possible phonetic alignments. This framework is inspired by the recently proposed minimum phone error (MPE) training approach and the minimum Bayes risk decoding algorithm for automatic speech recognition. To evaluate the proposed MBE framework, we conduct automatic phonetic segmentation experiments on the TIMIT acoustic-phonetic continuous speech corpus. MBE segmentation with MBE-trained models can identify 80.53% of human-labeled phone boundaries within a tolerance of 10 ms, compared to 71.10% identified by conventional ML segmentation with ML-trained models. Moreover, by using the MBE framework, only 7.15% of automatically labeled phone boundaries have errors larger than 20 ms.

Keywords: automatic phonetic segmentation, minimum boundary error, discriminative training, minimum Bayes risk.

1 Introduction

Many areas of speech technology exploit automatic learning methodologies that rely on large well-labeled corpora. Phoneme level transcription is especially important for fundamental speech research. In recent years, increased attention has been paid to data-driven, concatenation-based TTS synthesis because its output is more natural and has a high degree of fluency. Both the development of concatenative acoustic unit inventories and the statistical training of data-driven prosodic models require a speech database that is precisely segmented. In the past, the speech synthesis has relied on manually segmented corpora; however, such corpora are extremely hard to obtain, since labeling by hand is time consuming and costly. In speech recognition tasks, though the use of Hidden Markov Models (HMMs) has made finding precise phonetic boundaries unnecessary, it is believed that speech recognition would benefit from more precise segmentation in training and recognition.

To reduce the manual effort and accelerate the labeling process, many attempts have been made to utilize automatic phonetic segmentation approaches to provide initial phonetic segmentation for subsequent manual segmentation and verification, e.g.,

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 399 – 409, 2006.

dynamic time warping (DTW) [1], methods that utilize specific features and algorithms [2], HMM-based Viterbi forced alignment [3], and two-stage approaches [4].

The most popular method of automatic phonetic segmentation is to adapt an HMM-based phonetic recognizer to align a phonetic transcription with a speech utterance. Empirically, phone boundaries obtained in this way should contain few serious errors, since HMMs generally capture the acoustic properties of phones; however, small errors are inevitable because HMMs are not sensitive enough to detect changes between adjacent phones [4]. To improve the discriminability of HMMs for automatic phonetic segmentation, we proposed using a discriminative criterion, called the minimum boundary error (MBE), for model training in our previous work [5]. In this paper, the MBE criterion is extended to the segmentation stage, i.e., we propose an MBE forced alignment to replace the conventional maximum likelihood (ML) forced alignment. The superiority of the MBE framework over the conventional ML framework for automatic phonetic segmentation is verified by experiments conducted on the TIMIT acoustic-phonetic continuous speech corpus.

The remainder of this paper is organized as follows. Section 2 reviews the methodology of the MBE discriminative training approach. In Section 3, we present the proposed MBE segmentation approach and discuss its relation to the minimum Bayes risk (MBR) criterion. The experiment results are detailed in Section 4. Finally, in Section 5, we present our conclusions and suggest some future research directions.

2 Minimum Boundary Error Training

Let $\mathbf{O}=\{O^1,..,O^R\}$ be a set of training observation sequences. The objective function for MBE training can then be defined as:

$$F_{MBE}=\sum_{r=1}^{R}\sum_{S_i^r\in\mathbf{\Phi}^r}P(S_i^r \mid O^r)ER(S_i^r,S_c^r)\,, \tag{1}$$

where $\mathbf{\Phi}^r$ is a set of possible phonetic alignments for the training observation utterance O^r ; S_i^r is one of the hypothesized alignments in $\mathbf{\Phi}^r$; $P(S_i^r \mid O^r)$ is the posterior probability of alignment S_i^r , given the training observation sequence O^r ; and $ER(S_i^r,S_c^r)$ denotes the "boundary error" of S_i^r compared with the manually labeled phonetic alignment S_c^r . For each training observation sequence O^r , F_{MBE} gives the weighted average boundary error of all hypothesized alignments. For simplicity, we assume the prior probability of alignment S_i^r is uniformly distributed, and the likelihood $p(O^r \mid S_i^r)$ of alignment S_i^r is governed by the acoustic model parameter set Λ . Therefore, Eq.(1) can be rewritten as:

$$F_{MBE}=\sum_{r=1}^{R}\sum_{S_i^r\in\mathbf{\Phi}^r}\frac{p_\Lambda(O^r \mid S_i^r)^\alpha}{\sum_{S_k^r\in\mathbf{\Phi}^r}p_\Lambda(O^r \mid S_k^r)^\alpha}ER(S_i^r,S_c^r)\,, \tag{2}$$

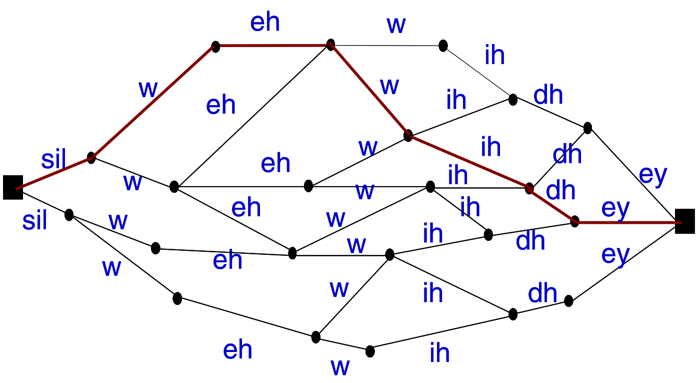


Fig. 1. An illustration of the phonetic lattice for the speech utterance: “Where were they?” The lattice can be generated by performing a beam search using some pruning techniques

where α is a scaling factor that prevents the denominator $\sum_{S_k^r \in \Phi^r} p_\Lambda(O^r \mid S_k^r)$ being dominated by only a few alignments. Accordingly, the optimal parameter set Λ^* can be estimated by minimizing the objective function defined in Eq.(2) as follows:

$$\Lambda^* = \arg\min_{\Lambda} \sum_{r=1}^{R} \sum_{S_i^r \in \Phi^r} \frac{p_\Lambda(O^r \mid S_i^r)^\alpha}{\sum_{S_k^r \in \Phi^r} p_\Lambda(O^r \mid S_k^r)^\alpha} ER(S_i^r, S_c^r) \cdot \tag{3}$$

The boundary error $ER(S_i^r, S_c^r)$ of the hypothesized alignment S_i^r can be calculated as the sum of the boundary errors of the individual phones in S_i^r, i.e.,

$$ER(S_i^r, S_c^r) = \sum_{n=1}^{N^r} er(q_n^i, q_n^c) \,, \tag{4}$$

where N^r is the number of total phones in O^r; q_n^i and q_n^c are the n-th phone in S_i^r and S_c^r, respectively; and $er(\cdot)$ is a phone boundary error function defined as,

$$er(q_n^i, q_n^c) = 0.5 \times (\left|s_n^i - s_n^c\right| + \left|e_n^i - e_n^c\right|) \,, \tag{5}$$

where s_n^i and e_n^i are the hypothesized start time and end time of phone q_n^i, respectively; and s_n^c and e_n^c correspond to the manually labeled start time and end time, respectively. Since $\mathbf{\Phi}^r$ contains a large number of hypothesized phonetic alignments, it is impractical to sum the boundary errors directly without first pruning some of the alignments. For efficiency, it is suggested that a reduced hypothesis space, such as an *N*-best list [6] or a lattice (or graph) [7], should be used. However, an *N*-best list often contains too much redundant information, e.g., two hypothesized alignments can be very similar. In contrast, as illustrated in Fig. 1, a phonetic lattice is more effective because it only stores alternative phone arcs on different segments of time marks and can easily generate a large number of distinct hypothesized phone alignments. Although it cannot be guaranteed that the phonetic alignments generated from a phonetic lattice will have higher probabilities than those not presented, we believe that the approximation will not affect the segmentation performance significantly. In this paper, we let $\mathbf{\Phi}^r_{\mathbf{Lat}}$ denote the set of possible phonetic alignments in the lattice for the training observation utterance O^r .

2.1 Objective Function Optimization and Update Formulae

Eq.(3) is a complex problem to solve, because there is no closed-form solution. In this paper, we adopt the Expectation Maximization (EM) algorithm to solve it. Since the EM algorithm maximizes the objective function, we reverse the sign of the objective function defined in Eq. (3) and re-formulate the optimization problem as,

$$\Lambda^* = \arg\max_{\Lambda} - \sum_{r=1}^{R} \sum_{S_i^r \in \mathbf{\Phi}^r} \frac{p_\Lambda(O^r \mid S_i^r)^\alpha}{\sum_{S_k^r \in \mathbf{\Phi}^r} p_\Lambda(O^r \mid S_k^r)^\alpha} ER(S_i^r, S_c^r) \ . \tag{6}$$

However, the EM algorithm can not be applied directly, because the objective function comprises rational functions [8]. The extended EM algorithm, which utilizes a weak-sense auxiliary function [9] and has been applied in the minimum phone error (MPE) discriminative training approach [10] for ASR, can be adapted to solve Eq.(6). The re-estimation formulae for the mean vector μ_m and the diagonal covariance matrix Σ_m of a given Gaussian mixture m thus derived can be expressed, respectively, as:

$$\mu_m = \frac{\theta_m^{MBE}(O) + D_m \overline{\mu}_m}{\gamma_m^{MBE} + D_m} , \tag{7}$$

and

$$\Sigma_m = \frac{\theta_m^{MBE}(O^2) + D_m \left\lfloor \overline{\Sigma}_m + \overline{\mu}_m \overline{\mu}_m^T \right\rfloor}{\gamma_m^{MBE} + D_m} - \mu_m \mu_m^T \ . \tag{8}$$

In Eqs. (7) and (8), D_m is a per-mixture level control constant that ensures all the variance updates are positive; $\overline{\mu}_m$ and $\overline{\Sigma}_m$ are the current mean vector and covariance matrix, respectively; and $\theta_m^{MBE}(O)$, $\theta_m^{MBE}(O^2)$, and γ_m^{MBE} are statistics defined, respectively, as:

$$\theta_m^{MBE}(O) = \sum_r \sum_{q \in \mathbf{\Phi}^r_{\mathbf{Lat}}} \sum_{t=s_q}^{t=e_q} \gamma_q^{r\,MBE} \gamma_{qm}^r(t) o_r(t) \ , \tag{9}$$

$$\theta_m^{MBE}(O^2) = \sum_r \sum_{q \in \mathbf{\Phi}^r_{\mathbf{Lat}}} \sum_{t=s_q}^{t=e_q} \gamma_q^{r\,MBE} \gamma_{qm}^r(t) o_r(t) o_r(t)^T \ , \tag{10}$$

and

$$\gamma_m^{MBE} = \sum_r \sum_{q \in \Phi_{\mathbf{Lat}}^r} \sum_{t=s_q}^{t=e_q} \gamma_q^{r\,MBE} \gamma_{qm}^r(t) \,. \tag{11}$$

In Eqs. (9), (10), and (11), $\gamma_{qm}^r(t)$ is the occupation probability of mixture m on q, $o_r(t)$ is the observation vector at time t, and $\gamma_q^{r\,MBE}$ is computed by

$$\gamma_q^{r\,MBE} = \gamma_q^r \left(\eta_{avg}^r - \eta_q^r \right), \tag{12}$$

where γ_q^r is the occupation probability of phone arc q, also referred to as its posterior probability; η_{avg}^r is the weighted average boundary error of all the hypothesized alignments in the lattice; and η_q^r is the weighted average boundary error of the hypothesized alignments in the lattice that contain arc q. Note that the term $\eta_{avg}^r - \eta_q^r$ reflects the difference between the weighted average boundary error of all the alignments in the lattice and that of the alignments containing arc q . When η_{avg}^r equals η_q^r, phone arc q makes no contribution to MBE training. However, when η_{avg}^r is larger than η_q^r, i.e., phone arc q generates fewer errors than the average, then q makes a positive contribution. Conversely, if η_{avg}^r is smaller than η_q^r, q makes a negative contribution. The discriminative ability of the MBE training approach is thus demonstrated. γ_q^r, η_{avg}^r, and η_q^r are computed by

$$\gamma_q^r = \frac{\sum_{S_i \in \Phi_{\mathbf{Lat}}^r, q \in S_i} p_{\overline{\Lambda}}(O_r \mid S_i)^\alpha}{\sum_{S_k \in \Phi_{\mathbf{Lat}}^r} p_{\overline{\Lambda}}(O_r \mid S_k)^\alpha}, \tag{13}$$

$$\eta_{avg}^r = \frac{\sum_{S_i \in \Phi_{\mathbf{Lat}}^r} p_{\overline{\Lambda}}(O_r \mid S_i)^\alpha ER(S_i)}{\sum_{S_k \in \Phi_{\mathbf{Lat}}^r} p_{\overline{\Lambda}}(O_r \mid S_k)^\alpha}, \tag{14}$$

and

$$\eta_q^r = \frac{\sum_{S_i \in \Phi_{\mathbf{Lat}}^r, q \in S_i} p_{\overline{\Lambda}}(O_r \mid S_i)^\alpha ER(S_i)}{\sum_{S_k \in \Phi_{\mathbf{Lat}}^r, q \in S_k} p_{\overline{\Lambda}}(O_r \mid S_k)^\alpha}, \tag{15}$$

respectively, where $\overline{\Lambda}$ is the current set of parameters. The above three quantities can be calculated efficiently by applying dynamic programming to the lattice.

2.2 I-Smoothing Update

To improve the generality of MBE training, the I-smoothing technique [10] is employed to provide better parameter estimates. This technique can be regarded as interpolating the MBE and ML auxiliary functions according to the amount of data available for each Gaussian mixture. The updates for the mean vector μ_m and the diagonal covariance matrix Σ_m thus become:

$$\mu_m = \frac{\theta_m^{MBE}(O) + D_m\overline{\mu}_m + \frac{\tau_m}{\gamma_m^{ML}}\theta_m^{ML}(O)}{\gamma_m^{MBE} + D_m + \tau_m}, \tag{16}$$

and

$$\Sigma_m = \frac{\theta_m^{MBE} + D_m\left[\overline{\Sigma}_m + \overline{\mu}_m\overline{\mu}_m^T\right] + \frac{\tau_m}{\gamma_m^{ML}}\theta_m^{ML}(O^2)}{\gamma_m^{MBE} + D_m + \tau_m} - \mu_m\mu_m^T, \tag{17}$$

respectively, where τ_m is also a per-mixture level control constant; and γ_m^{ML}, $\theta_m^{ML}(O)$, and $\theta_m^{ML}(O^2)$ are computed by

$$\gamma_m^{ML} = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\gamma_m^{r\ ML}(t), \tag{18}$$

$$\theta_m^{ML}(O) = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\gamma_m^{r\ ML}(t)o_r(t), \tag{19}$$

and

$$\theta_m^{ML}(O^2) = \sum_{r=1}^{R}\sum_{t=1}^{T_r}\gamma_m^{r\ ML}(t)o_r(t)o_r(t)^T, \tag{20}$$

respectively. In Eqs. (18), (19), and (20), T_r is the frame number of O_r, and $\gamma_m^{r\ ML}(t)$ is the maximum likelihood occupation probability of the Gaussian mixture *m*.

3 Minimum Boundary Error Segmentation

The proposed MBE forced alignment approach is a promising realization of the *Minimum Bayes-Risk* (MBR) classifier for the automatic phonetic segmentation task. The latter can be considered as taking an action, $\alpha_S(O)$, to identify a certain alignment, S, from all the various phonetic alignments of a given utterance O. Let function $L(S, S_c)$ be the loss incurred when the action $\alpha_S(O)$ is taken, given that the true (or reference) alignment is S_c. During the classification stage, we do not know the true alignment in advance, i.e., any arbitrary alignment S_j could be true. Suppose the distribution $P(S_j \mid O)$ is known, then the conditional risk of taking the action $\alpha_S(O)$ is given by:

$$R(\alpha_S \mid O) = \sum_{S_j} L(S, S_j) P(S_j \mid O) . \tag{21}$$

The MBR classifier is designed to select the action whose conditional risk, $R(\alpha_S \mid O)$, is minimal, i.e., the best alignment based on the MBR criterion can be found by

$$S^* = \arg\min_S R(\alpha_S \mid O) = \arg\min_S \sum_{S_j \in \mathbf{\Phi}} L(S, S_j) P(S_j \mid O) . \tag{22}$$

When the symmetrical zero-one function,

$$L(S, S_j) = \begin{cases} 0, & S = S_j \\ 1, & S \neq S_j \end{cases} , \tag{23}$$

is selected as the loss function, and it is assumed that the prior probability of alignment S_j is uniformly distributed, the MBR classifier is equivalent to the conventional forced-alignment method, which picks the alignment with the maximal likelihood, i.e.,

$$\begin{aligned} S^* &= \arg\min_S \sum_{S_j \in \mathbf{\Phi}} L(S, S_j) P(S_j \mid O) \\ &= \arg\min_S \sum_{S_j \in \mathbf{\Phi}, S_j \neq S} P(S_j \mid O) \\ &= \arg\min_S (1 - P(S \mid O)) \\ &= \arg\max_S P(O \mid S) \end{aligned} . \tag{24}$$

It is clear from Eq. (23) that the zero-one loss function assigns no loss when $S = S_j$, but assigns a uniform loss of one to the alignments $S \neq S_j$ no matter how different they are from S_j. Thus, such a loss function causes all incorrectly hypothesized alignments to be regarded as having the same segmentation risk, which is obviously inconsistent with our preference for alignments with fewer errors in an automatic segmentation task.

In our approach, the loss function is replaced by the boundary error function, defined in Eq.(4), to match the goal of minimizing the boundary error. Consequently, the MBR forced alignment approach becomes the MBE forced alignment approach, defined as:

$$\begin{aligned} S^* &= \arg\min_S \sum_{S_j \in \mathbf{\Phi}} ER(S, S_j) P(S_j \mid O) \\ &= \arg\min_S \sum_{S_j \in \mathbf{\Phi}} \sum_{n=1}^{N} er(q_n, q_n^j) P(S_j \mid O) \end{aligned} , \tag{25}$$

where N is the number of phones in utterance O; and q_n and q_n^j are the n-th phone in the alignments S and S_j, respectively.

To simplify the implementation, we restrict the hypothesized space $\mathbf{\Phi}$ to $\mathbf{\Phi}_{\mathbf{Lat}}$, the set of alignments constructed from the phone lattice shown in Fig. 1, which can be generated by a conventional beam search. Accordingly, Eq. (25) can be re-formulated as:

$$\begin{aligned} S^* &= \arg\min_{S} \sum_{S_j \in \mathbf{\Phi}_{\mathbf{Lat}}} \sum_{n=1}^{N} P(S_j \mid O) er(q_n, q_n^j) \\ &= \arg\min_{S} \sum_{n=1}^{N} \sum_{S_j \in \mathbf{\Phi}_{\mathbf{Lat}}} P(S_j \mid O) er(q_n, q_n^j) \end{aligned} . \quad (26)$$

Let the ***cut*** $\mathbf{C}_n$ be the set of phone arcs of the n-th phone in the utterance. For example, in Fig. 1, there are four phone arcs for the second phone, "w", in $\mathbf{C}_2$ and six phone arcs for the third phone, "eh", in $\mathbf{C}_3$. From the figure, it is obvious that each alignment in $\mathbf{\Phi}_{\mathbf{Lat}}$ will pass a single phone arc in each ***cut*** $\mathbf{C}_n$, n=1,2,…,N. According to this observation, Eq. (26) can be rewritten as:

$$S^* = \arg\min_{S} \sum_{n=1}^{N} \sum_{q_{n,m} \in \mathbf{C}_n} \sum_{\{S_j \in \mathbf{\Phi}_{\mathbf{Lat}} \mid q_{n,m} \in S_j\}} P(S_j \mid O) er(q_n, q_{n,m}) , \quad (27)$$

where $q_{n,m}$ is the m-th phone arc in $\mathbf{C}_n$. Because $\sum_{\{S_j \in \mathbf{\Phi}_{\mathbf{Lat}} \mid q_{n,m} \in S_j\}} P(S_j \mid O)$ in Eq. (27) is equivalent to the posterior probability of $q_{n,m}$ given the utterance O, denoted as $\gamma_{q_{n,m}}$ hereafter, the probability can be easily calculated by applying a forward-backward algorithm to the lattice. As a result, Eq. (27) can be rewritten as:

$$S^* = \arg\min_{S} \sum_{n=1}^{N} \sum_{q_{n,m} \in \mathbf{C}_n} \gamma_{q_{n,m}} er(q_n, q_{n,m}) . \quad (28)$$

In this way, MBE forced alignment can be efficiently conducted on the phone lattice by performing Viterbi search.

4 Experiments

4.1 Experiment Setup

TIMIT (The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus) [11], a well-known read speech corpus with manual acoustic phonetic labeling, has been widely used to evaluate automatic speech recognition and phonetic segmentation techniques. TIMIT contains a total of 6,300 sentences spoken by 630 speakers from eight major dialect regions in the United States; each speaker utters 10 sentences. The TIMIT suggested training and testing sets contain 462 and 168 speakers, respectively. We discard utterances with phones shorter than 10 ms. The resulting training set contains 4,546 sentences, with a total length of 3.87 hours, while the test set contains 1,646 sentences, with a total length of 1.41 hours.

The acoustic models consist of 50 context-independent phone models, each represented by a 3-state continuous density HMM (CDHMM) with a left-to-right topology.

Each frame of the speech data is represented by a 39-dimensional feature vector comprised of 12 MFCCs and log energy, plus their first and second differences. The frame width is 20 ms and the frame shift is 5 ms. Utterance-based cepstral variance normalization (CVN) is applied to all the training and test speech utterances.

4.2 Experiment Results

The acoustic models were first trained on the training utterances according to human-labeled phonetic transcriptions and boundaries by the Baum-Welch algorithm using the ML criterion. Then, the MBE discriminative training approach was applied to further manipulate the models. The scaling factor α in Eq.(2) was empirically set to 0.1 and the I-smoothing control constant τ_m in Eqs.(16) and (17) was set to 20 for all mixtures. The results are shown in Fig. 2. In the figure, the line with triangles indicates the expected FER (frame error rate) calculated at each iteration of the training process. Clearly, the descending trend satisfies the training criterion. The line with diamonds and the line with rectangles represent the FER results of the training (inside test) and test sets, respectively. We observe that the ML-trained acoustic models (at the 0th iteration) yield an FER of 10.31% and 11.77% for the training set and test set respectively. In contrast, after 10 iterations, the MBE-trained acoustic models yield an FER of 6.88% and 9.25%, respectively. The MBE discriminative training approach achieves a relative FER reduction of 33.27% on the training set and 21.41% on the test set. The results clearly demonstrate that the MBE discriminative training approach performs very well and can enhance the performance of the acoustic models initially trained by the ML criterion.

Table 1 shows the percentage of phone boundaries correctly placed within different tolerances with respect to their associated manually-labeled phone boundaries. The experiment was conducted on the test set. From rows 2 and 3 of Table 1, we observe that the MBE-trained models significantly outperform the ML-trained models. Clearly, the MBE training is particularly effective in correcting boundary errors in the proximity of manually labeled positions. Comparing the results in rows 2 and 4, we also observe that MBE segmentation outperforms ML segmentation, though the improvement is not as significant as that of the MBE-trained models over the ML-trained models. This is because, MBE segmentation, like conventional ML segmentation, is still deficient in the knowledge of true posterior distribution, even though the MBE criterion accords with the objective of minimizing boundary errors very well. The 5th row of Table 1 shows the results obtained when the complete MBE framework, including MBE training and MBE segmentation, was applied. We observe that these results are superior to those achieved when either the MBE training or the MBE segmentation was applied alone. The last row of Table 1 shows the absolute improvements achieved by the MBE framework over the conventional ML framework. The proposed MBE framework can identify 80.53% of human-labeled phone boundaries within a tolerance of 10 ms, compared to 71.10% identified by the conventional ML framework. Moreover, by using the MBE framework, only 7.15% of automatically labeled phone boundaries have errors larger than 20 ms.

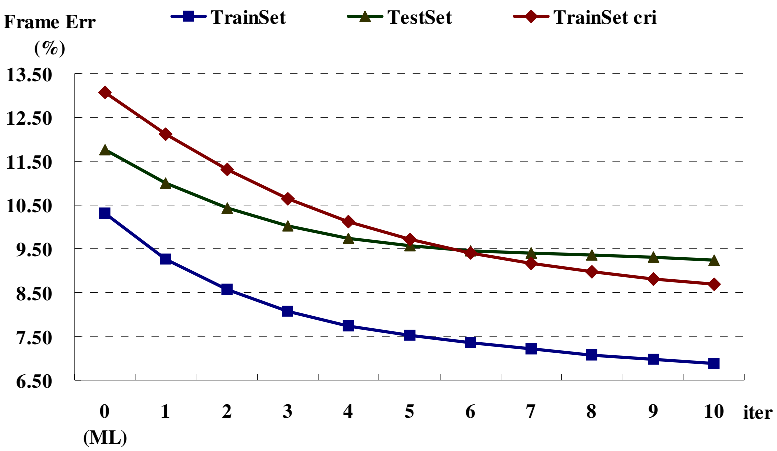


Fig. 2. The phonetic segmentation results (FER) for the models trained according to ML and MBE criteria, respectively

Table 1. The percentage of phone boundaries correctly placed within different tolerances with respect to their associated manually labeled phone boundaries

Criterion		Mean Boundary Distance	%Correct marks (distance ≦ tolerance)					
Training	Segmentation		≦5ms	≦10ms	≦15ms	≦20ms	≦25ms	≦30ms
ML	ML	9.83 ms	46.69	71.10	83.14	88.94	92.32	94.52
ML+MBE	ML	7.82 ms	58.48	79.75	88.16	92.11	94.49	96.11
ML	MBE	8.95 ms	49.86	74.25	85.38	90.61	93.75	95.67
ML+MBE	MBE	7.49 ms	58.73	80.53	88.97	92.85	95.16	96.64
absolute improvement (ML+MBE,MBE) vs. (ML, ML)		2.34 ms	12.04	9.43	5.83	3.91	2.84	2.12

5 Conclusions and Future Work

In this paper, we have explored the use of the minimum boundary error (MBE) criterion in the discriminative training of acoustic models as well as minimum risk segmentation for automatic phonetic segmentation. The underlying characteristics of the MBE training and segmentation framework have been investigated, and its superiority over conventional ML training and segmentation has been verified by experiments. Naturally, the more accurate phonetic segmentation obtained by the MBE framework is very useful for subsequent manual verification or further boundary refinement using other techniques. It is worth mentioning that the MBE training method is not difficult to implement; in particular, minimum phone error training has been included in HTK.

In HMM-based automatic phonetic segmentation and speech recognition tasks, duration control is an important issue that must be addressed. We tried to apply the MBE criterion in duration model training, but there was no significant improvement found in our preliminary work. However, the issue warrants further study. On the other hand, well-labeled phonetic training corpora are very scarce. Therefore, the unsupervised MBE training approach is also under investigated. Moreover, in our current implementation, the phone boundary error function, defined in Eq.(5), is calculated in the time frame unit for efficiency. However, more accurate segmentation may be achieved by calculating boundary errors in actual time sample marks. In addition, we are applying the MBE training and segmentation framework to facilitate the phonetic labeling of a subset of speech utterances in MATBN (Mandarin across Taiwan – Broadcast News) database [12].

Acknowledgment. This work was funded by the National Science Council, Taiwan, under Grant: NSC94-2213-E-001-021.

References

1. Malfrere, F., Dutiot, T.: High-quality speech synthesis for phonetic speech segmentation. Proc. Fifth Eurospeech (1997) 2631-2634
2. van Santen, J., Sproat, R.: High accuracy automatic segmentation. Proc. Sixth Eurospeech (1999) 2809-2812
3. Brugnara, F., Falavigna, D., Omologo, M.: Automatic segmentation and labeling of speech based on Hidden Markov Models. Speech Communication, Vol. 12, Issue. 4 (1993) 357-370
4. Torre Toledano, D., Rodriguez Crespo, M. A., Escalada Sardina, J. G.: Try to mimic human segmentation of speech using HMM and fuzzy logic post-correction rules. Proc. Third ESCA/COCOSDA International Workshop on Speech Synthesis (1998) 1263-1266
5. Kuo, J.-W., Wang, H.-M.: Minimum Boundary Error Training for Automatic Phonetic Segmentation. Proc. Interspeech – ICSLP (2006)
6. Schwartz, R., Chow, Y.-L.: The N-best algorithms: an efficient and exact procedure for finding the N most likely sentence hypotheses. Proc. ICASSP, Vol. 1(1990) 81-84
7. Ortmanns, S., Ney, H., Aubert, X.: A word graph algorithm for large vocabulary continuous speech recognition. Computer Speech and Language, Vol. 11 (1997) 43-72
8. Gopalakrishnan, P., Kanevsky, D., Nádas, A., Nahamoo, D.: An inequality for rational functions with applications to some statistical estimation problems. IEEE Trans. Information Theory, Vol. 37 (1991) 107-113
9. Povey, D.: Discriminative Training for Large Vocabulary Speech Recognition. Ph.D. Dissertation, Peterhouse, University of Cambridge, July 2004
10. Povey, D., Woodland, P. C.: Minimum phone error and I-smoothing for improved discriminative training. Proc. ICASSP, Vol. 1 (2002) 105-108
11. Lamel, L., Kasel, R., Seneff, S.: Speech database development: design and analysis of the acoustic-phonetic corpus. Proc. DARPA Speech Recognition Workshop (1986) 100-109
12. Wang, H.-M., Chen, B., Kuo, J.-W., Cheng, S.-S.: MATBN: A Mandarin Chinese Broadcast News Corpus. International Journal of Computational Linguistics & Chinese Language Processing, Vol. 10, No. 2, June (2005) 219-236

Advances in Mandarin Broadcast Speech Transcription at IBM Under the DARPA GALE Program

Yong Qin[1], Qin Shi[1], Yi Y. Liu[1], Hagai Aronowitz[2], Stephen M. Chu[2],
Hong-Kwang Kuo[2], and Geoffrey Zweig[2]

[1] IBM China Research Lab, Beijing 100094
{qinyong, shiqin, liuyyi}@cn.ibm.com
[2] IBM T. J. Watson Research Center, Yorktown Heights, New York 10598, U.S.A
{haronow, schu, hkuo, gzweig}@us.ibm.com

Abstract. This paper describes the technical and system building advances in the automatic transcription of Mandarin broadcast speech made at IBM in the first year of the DARPA GALE program. In particular, we discuss the application of *minimum phone error* (MPE) discriminative training and a new topic-adaptive language modeling technique. We present results on both the RT04 evaluation data and two larger community-defined test sets designed to cover both the broadcast news and the broadcast conversation domain. It is shown that with the described advances, the new transcription system achieves a 26.3% relative reduction in character error rate over our previous best-performing system, and is competitive with published numbers on these datasets. The results are further analyzed to give a comprehensive account of the relationship between the errors and the properties of the test data.

Keywords: discriminative training, topic-adaptive language model, mandarin, broadcast news, broadcast conversation.

1 Introduction

This paper describes Mandarin speech recognition technology developed at IBM for the U.S. Defense Advanced Research Projects Agency (DARPA) Global Autonomous Language Exploitation (GALE) program. The overall goal of this program is to extract information from publicly available broadcast sources in multiple languages, and to make it accessible to monolingual English speakers. In order to accomplish this, the program has several major components: *speech recognition*, *machine translation*, and *question answering* (formally termed *distillation*).

In the IBM approach implemented in 2006, broadcasts are sequentially processed with these technologies: first, speech recognition is used to create a textual representation of the source language speech; second, machine translation is used to convert this to an English language representation; and thirdly, question answering technology is used to answer queries like "Tell me the mutual acquaintances of [*person*] and [*person*]" or "Tell me [*person*]'s relationship to [*organization*]." While IBM has developed systems for GALE's two target languages, Arabic and Mandarin, and has

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 410 – 421, 2006.

participated in all three activities, this paper focuses solely on the Mandarin language automatic speech recognition (ASR) component.

The GALE program focuses on two types of broadcast audio: broadcast news – which was a focus of attention in the previous DARPA Effective Affordable Reusable Speech-to-text (EARS) and HUB-4 programs – and broadcast conversations. The study of broadcast conversations is relatively new to the speech recognition community, and the material is more challenging than broadcast news shows. Whereas broadcast news material usually includes a large amount of carefully enunciated speech from anchor speakers and trained reporters, broadcast conversations are more unplanned and spontaneous in nature, with the associated problems of spontaneous speech: pronunciation variability, rate-of-speech variability, mistakes, corrections, and other disfluencies. Further, in Chinese ASR, we are faced with the problem of an ambiguous segmentation of characters into words – a problem not seen in languages such as English or Arabic.

This paper will describe our Mandarin recognition work from the system-building perspective, and present a detailed error analysis. The main contributions of the paper are: (*a*) the presentation and validation of effective Mandarin system architecture, (*b*) an adaptive language modeling technique, and (*c*) a careful error analysis.

The error analysis in particular indicates that: (1) Style of speech, broadcasting network and gender are the most important attributes and can vary *character error rate* (CER) from 5.7% to 31.5%; (2) Telephone speech, and speech plus noise or music cause an absolute degradation of 2-3% each; (3) Rate-of-speech (characters per second) is an important attribute and can degrade CER up to 60%; and (4) Short speakers are bad speakers – speakers who talk less than 30 seconds per show have a 77% relative higher CER. We note that this is not due to a lack of adaptation data, as we find low error rates for small amounts of speech sampled from "long" speakers.

Also of interest and somewhat unexpected are our gains from discriminative training, which we observe to be relatively large compared to those we see in English and Arabic.

The remainder of this paper is organized as follows. In Section 2, we present our system architecture. This architecture amalgamates techniques used previously in English [1], as well as extending it with a novel adaptive language modeling technique. In Section 3, we describe the specifics of our Mandarin system, including the training data and system size. Section 4 presents experimental results on broadcast news and broadcast conversation test sets. In Section 5, we present our error analysis, followed by conclusions in Section 6.

2 System Architecture

The IBM GALE Mandarin broadcast speech transcription system is composed of three main stages, speech segmentation/speaker clustering, speaker independent (SI) decoding, and speaker adapted (SA) decoding. A system diagram is shown in Fig. 1. In this section, we describe the various components of the system.

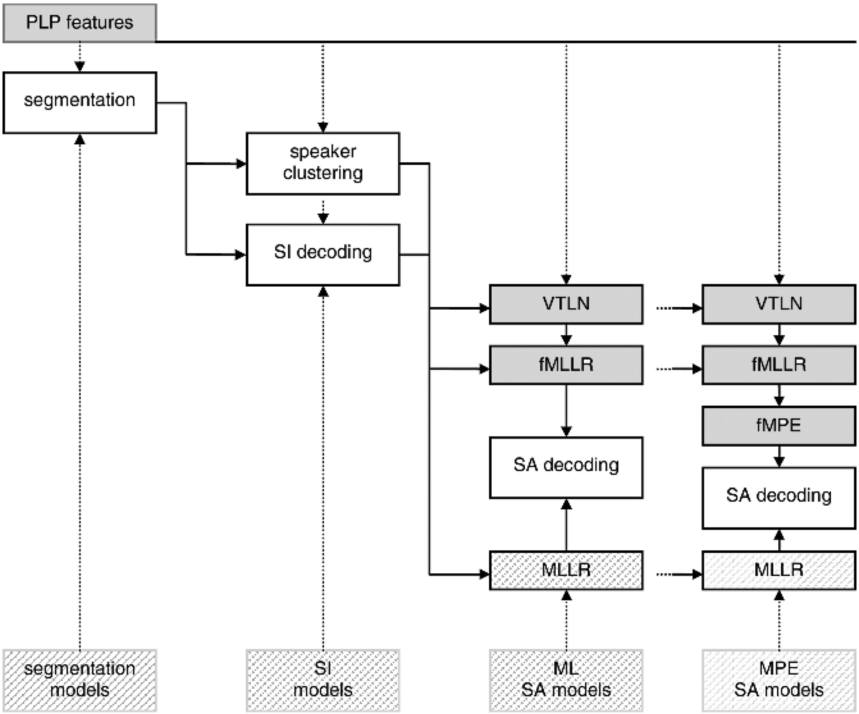


Fig. 1. The IBM Mandarin broadcast speech transcription system consists of speech detection/segmentation, speaker clustering, speaker independent decoding, and speaker adapted decoding. In speaker adapted decoding, both feature and model space adaptations are applied. Models and transforms discriminatively that are trained using *minimum phone error* training provide further refinement in acoustic modeling.

2.1 Front-End Processing

The basic features used for segmentation and recognition are *perceptual linear prediction* (PLP) features. Feature mean normalization is applied as follows: in segmentation and speaker clustering, the mean of the entire session is computed and subtracted; for SI decoding, speaker-level mean normalization is performed based on the speaker clustering output; and at SA stage, the features are mean and variance normalized for each speaker. Consecutive feature frames are spliced and then projected back to a lower dimensional space using *linear discriminant analysis* (LDA), which is followed by a *maximum likelihood linear transform* (MLLT) [2] step to further condition the feature space for diagonal covariance Gaussian densities.

2.2 Segmentation and Clustering

The segmentation step uses an HMM-based classifier. The speech and non-speech segments are each modeled by a five-state, left-to-right HMM with no skip states. The output distributions are tied across all states within the HMM, and are specified by a mixture of Gaussian densities with diagonal covariance matrices.

After segmentation, the frames classified as non-speech are discarded, and the remaining segments are put through the clustering procedure to give speaker hypotheses. The clustering algorithm models each segment with a single Gaussian density and clusters them into a pre-specified number of clusters using K-means.

Note that in the broadcast scenario, it is common to observe recurring speakers in different recording sessions, e.g., the anchors of a news program. Therefore, it is possible to create speaker clusters beyond the immediate broadcast session. Nevertheless, in the scope of this paper, we shall restrict the speaker clustering procedure to a per session basis.

2.3 SI Models

The system uses a tone-specific phone set with 162 phonemes. Phones are represented as three-state, left-to-right HMMs. With the exception of silence and noise, the HMM states are context-dependent conditioned on quinphone context covering both past and future words. The context-dependent states are clustered into equivalence classes using a decision tree.

Emission distributions of the states are modeled using mixtures of diagonal-covariance Gaussian densities. The allocation of mixture component to a given state is a function of the number of frames aligned to that state in the training data. Maximum likelihood (ML) training is initialized with state-level alignment of the training data given by an existing system. A mixture-splitting algorithm iteratively grows the acoustic model from one component per state to its full size. One iteration of Viterbi training on word graphs is applied at the end.

2.4 SA Models

The SA acoustic models share the same basic topology with the SI model. For speaker adaptation, a model-space method, *maximum likelihood linear regression* (MLLR), and two feature-space methods, *vocal tract length normalization* (VTLN) [3] and *feature-space* MLLR (fMLLR) [4], are used in the baseline system.

An eight-level binary regression tree is used for MLLR, which is grown by successively splitting the nodes from the top using soft K-means algorithm. The VTLN frequency warping consists of a pool of 21 piecewise linear functions, or warping factors. In decoding, a warping factor is chosen such that it maximizes the likelihood of the observations given a voice model built on static features with full covariance Gaussian densities.

In addition to the speaker adaptation procedures, the improved Mandarin transcription system also employs the discriminately trained *minimum phone error* (MPE) [5] models and the recently developed *feature-space* MPE (fMPE) [6] transform. Experiments show that these discriminative algorithms give a significant improvement to recognition performance. The results are presented in Section 4. Here we briefly review the basic formulation.

2.5 MPE/fMPE Formulation

The objective function of MPE [5] is an average of the transcription accuracies of all possible sentences s, weighted by the probability of s given the model:

$$\Phi_{MPE}(\lambda) = \sum_{r=1}^{R}\sum_{s} P_{\lambda}(s \mid \mathrm{O}_r)\mathrm{A}(s, s_r) \tag{1}$$

where $P_{\lambda}(s \mid \mathrm{O}_r)$ is defined as the scaled posterior sentence probability of the hypothesized sentence s:

$$\frac{p_{\lambda}(\mathrm{O}_r \mid s)^{\kappa} P(s)^{\nu}}{\sum_{u} p_{\lambda}(\mathrm{O}_r \mid u)^{\kappa} P(u)^{\nu}}, \tag{2}$$

λ denotes the model parameters, κ and ν are scaling factors, and O_r the acoustics of the r'th utterance. The function $\mathrm{A}(s, s_r)$ is a "raw phone accuracy" of s given the reference s_r, which equals the number of phones in the reference minus the number of phone errors made in sentence r.

The objective function of fMPE is the same as that of MPE. In fMPE [6], the observation of each time frame $\mathbf{x}_t$ is first converted to a high-dimensional feature vector $\mathbf{h}_t$ by taking posteriors of Gaussians, which is then projected back to the original lower dimensional space using a global discriminatively trained transform. The resulting vector and the original observation are added to give the new feature vector $\mathbf{y}_t$:

$$\mathbf{y}_t = \mathbf{x}_t + \mathbf{M}\mathbf{h}_t \tag{3}$$

Thus, the fMPE training constitutes the learning of projection matrix **M** using the MPE objective function.

Previous experiments have indicated that combing fMPE with model space discriminative training can further improve recognition performance [6]. In practice, we first obtain the fMPE transform using ML trained acoustic models, and then a new discriminately trained model is built upon the fMPE features using MPE.

2.6 Language Modeling

The language models (LM) considered in this work are interpolated back-off 4-gram models smoothed using modified Kneser-Ney smoothing [7]. The interpolation weights are chosen to optimize the perplexity of a held-out data set.

In addition to the basic language models, we also developed a topic-adaptive language modeling technique using a multi-class *support vector machines* (SVM) -based topic classifier[1]. The topics are organized as a manually constructed tree with 98 leaf nodes. To train the classifier, more than 20,000 Chinese news articles covering a wide range of topics are collected and annotated. The raw feature representing each training sample is a vector of terms given by our Mandarin text segmenter. A SVM is then trained to map from these feature vectors to topics. To reduce nuisance features, words occurring in less than three documents are omitted. The overall classification accuracy of the topic classifier as measured by the F_1 measure is 0.8. An on-topic LM is trained for each of the 98 classes.

[1] The text classification tool is developed by Li Zhang at IBM China Research Lab.

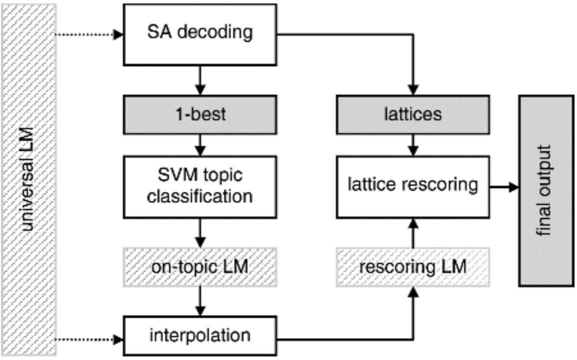


Fig. 2. Topic adaptation is carried out through lattice rescoring with an LM interpolated from the universal LM and a topic-specific LM. Topic classification is based on the 1-best word hypothesis given by the SA decoding output.

In decoding, the basic universal LM is first used to generate a word lattice and the 1-best hypothesis. The 1-best hypothesis is subsequently used for topic classification. Note that the change of topic occurs frequently in broadcast materials. Therefore, the classification is performed at the utterance level. Base on the classification result, an on-topic LM is selected from the 98 pre-trained LMs and interpolated with the universal LM. The resulting LM is used to rescore the lattices generated earlier to give the final recognition output. The process is shown in Fig 2.

3 System Building

3.1 Training Data

The majority of our acoustic modeling data was obtained from the Linguistic Data Consortium (http://www.ldc.upenn.edu), as was the bulk of our language modeling data. The acoustic Modeling data is summarized in Table 1. A relatively small amount consists of broadcasts of news shows transcribed internally at IBM, and labeled “Satellite Data” below. From the data sources listed 550 hours were used to train our acoustic models, based on data that aligned to the transcripts using a set of boot models.

Table 1. Acoustic modeling data (with full transcripts)

Corpora		BN (Hours)	BC (Hours)
LDC1998T24	(HUB4)	30.0	--
LDC2005E63	(GALE kickoff)	--	25.0
LDC2006E23	(GALE Y1)	74.9	72.7
LDC2005S11	(TDT4)	62.8	--
LDC2005E82	(Y1Q1)	50.2	7.58
LDC2006E33	(Y1Q2)	136.6	76.1
SATELLITE		50.0	--

Our language model was built from all the acoustic transcripts, and additional text data that were used solely for language modeling purposes. This data is listed in Table 2.

Table 2. Language modeling Data

Copora	Type	Number of words
LDC1995T13	Newswire	116M
LDC2000T52	Newswire	10.1M
LDC2003E03	News	1.4M
LDC2004E41	Newswire	17.1M
LDC2005T14	Newswire	245M
LDC2001T52	BN	4.7M
LDC2001T58	BN	3.1M
LDC2005E82 LDC2006E33	Blog & Newsgroup	17.2M
SRI Web 20060522	Web	183M (characters)
SRI Web 20060608	Web	5M (characters)

3.2 System Description

The 16 KHz input signal is coded using 13-dememsional PLP features with a 25ms window and 10ms frame-shift. Nine consecutive frames are spliced and projected to 40 dimensions using LDA. The SI acoustic model has 10K quinphone states modeled by 150K Gaussian densities. The SA model uses a larger tree with 15K states and 300K Gaussians.

In addition to fully transcribed data, the training corpora also contain broadcast recordings with only closed captioning text. To take advantage of these data, *lightly supervised training* is applied.

The method relies on an automatic way to select reliable segments from the available data. First, we use the closed caption to build a *biased* LM. Then, the *biased* LM in conjunction with the existing acoustic model is used to decode the corresponding audio. The decoded text is aligned with the closed caption, and a segment is discarded unless it satisfies the following two criteria: (a) the longest successful alignment is more than three words; and (b) the decoding output ends on a silence word. The surviving data are deemed reliable and used for acoustic model training. This method is similar to those presented in [9] and [10].

It is observed that for broadcast news (BN) content, 55% of the closed caption data are eventually used in training, whereas for broadcast conversations (BC), only 21% survived the filtering process. In total, lightly supervised training increases the training set by 143 hours.

Our language model consists of an interpolation of eleven distinct models built from subsets of the training data. Subsets are listed in Table 3. A held-out set with 31K words (61% BC, 39% BN) is used to determine the interpolation weights. The resulting LM has 6.1M n-grams, and perplexities of 735, 536, and 980 on RT04, 2006E10, and devo5bcm respectively.

Table 3. Training subsets and statistics of the 11 LMs

LM	LDC catalog Number: [Sets Used]	# of words	# of n-grams	RT04 PPL	2006E10 PPL	dev05bcm PPL	Data Category
1	2005T14:TDT2-4, 2004E41, 2000T52	326M	61.7M	1080	654.8	2773	newspaper
2	2005T14: 1991~1999 Taiwan data	180M	33.7M	1981	1407	3571	newspaper (Taiwan)
3	2005T14: 2000~2004 Taiwan data	41.6M	62.0M	1675	1237	3359	newspaper (Taiwan)
4	(IBM Chinese Web News Collection)	133M	55.9M	1129	808	2174	web news
5	1998S73_T24, 2005E61-63: BN data	5.93M	10.0M	1415	1237	3359	BC
6	GALE Y1Q1Q2: BN, NTDTVWEB, RT-03 BN training text, Satellite	4.25M	7.50M	1433	1219	1608	BN
7	95T13	124M	119M	1545	965.8	3479	newspaper
8	GALE Y1Q1, GALE Y1Q2	16.5M	17.1M	1563	1378	2293	weblog, newsgroup
9	FOUO_SRIWebText.20060522	69.8M	98.4M	987.3	704.5	1412	web text
10	GALE Y1Q1Q2: BC force alignment	1.92M	3.21M	2803	2466	1567	BC
11	GALE Y1Q1Q2: BN force alignment	2.52M	3.87M	1739	1339	2020	BN

4 Experimental Results

Three test sets are used to evaluate the Mandarin broadcast transcription system. The first is the evaluation set from the Rich Transcription'04 (RT04) evaluation's Mandarin broadcast news task. It contains 61 minutes of data drawn from four BN recordings. The second test set, denoted "dev05bcm", contains five episodes of three BC programs. The total duration of this set is 3.5 hours. A third, 4.5-hour BN set "2006E10" is included to give more robust coverage of the BN content. The 2006E10 test set may be downloaded from the LDC, and includes RT04. The list of dev05bcm audio files was created at Cambridge University and distributed to GALE participants.

Recognition experiments on the three test sets are carried out following the pipeline shown in Fig. 1 in section 2. At the SA level, decoding using the ML acoustic model is done at after VTLN, after fMLLR, and after fMLLR to further understand the effect of each adaptation step on the Mandarin broadcast speech transpiration task. Except for VTLN decoding, the experiments are repeated using the MPE trained models and features. The recognition results are summarised in Table 4.

Table 4. Character error rates observed on the three test sets at different level of acoustic model refinement. The results indicate that discriminative training gives significant improvements in recognition performance.

System Build Level		RT04	dev05bcm	2006E10
SI:	--	19.4	28.3	19.6
SA:	VTLN	18.0	26.7	18.1
	+fMLLR	16.5	24.9	17.1
	+MLLR	15.7	24.3	16.7
	+fMPE+MPE+fMLLR	14.3	22.0	14.0
	+MLLR	13.7	21.3	13.8

As expected, the results show that BC data (dev05bcm) pose a greater challenge than the two BN sets. The results clearly confirm the effectiveness of the adaptive and discriminative acoustic modeling pipeline in the system. Furthermore, the overall trend of the CER as observed in each column is consistent across all three sets. In particular, we note that the MPE/fMPE algorithm gives a relatively large improvement to recognition performance on top of speaker adaptation. For instance, on 2006E10, discriminative training further reduces the CER by 2.9% absolute to 13.8% from the best ML models. Similarly, a 3.0% absolute reduction is achieved on the dev05bcm set. As a comparison, the MPE/fMPE gain observed in our Arabic broadcast transcription system is 2.1% absolute on the RT04 Arabic set.

To track the progress made in the GALE engagement, we compare the performance of the current system (06/2006) with our system at the end of 2005 (12/2005). The results are shown in Table 5. On RT04, a relative reduction in CER of 26.3% is observed. For reference, the best published numbers in the community on RT04 and dev05bcm are also listed [11].

Table 5. Comparing character error rates of the current system with the previous best-performing system and the best published results on the same test sets [11]

	System ID	RT04	dev05bcm	2006E10
SI:	12/2005	22.5	39.6	22.8
	06/2006	19.4	28.3	19.6
SA:	12/2005	18.6	34.5	20.0
	06/2006	13.7	21.3	13.8
Best published number		14.7	25.2	--

Finally, the topic-adaptive language modelling technique is evaluated by rescoring the SA lattices with (1) the LM that is topic-adapted to a given test utterance, and (2) an LM interpolated from the universal LM and a fixed set of eight topic-dependent LMs. On RT04, results show that the adaptive approach gives 0.4% absolute reduction in CER comparing with the non-adaptive counterpart.

5 Error Characterization

In this section, we aim to gain a better understanding of the Mandarin broadcast speech transcription task by analyzing the correlation between the error made by our system and the various attributes of the data. Unfortunately, the three LDC test sets used earlier lack the rich annotation required by such a study. Therefore, we use a dataset that has been carefully annotated at IBM for this part of the paper. The data are collected from the same program sources as the LDC sets, and are selected to have a comparable content composition. Six shows were recorded from the CCTV4 network and 4 from the Dragon network. The total duration of the dataset is 6 hours. In order to have attribute-homogenous segments for analysis, we used manually marked speaker boundaries and speaker identities.

5.1 Method and Results

Table 6 lists the attributes used for CER analysis. We investigated five categorical attributes: gender, style, network, speech quality, and channel; and two numerical attributes: amount of speech per speaker and character rate. Binary categorical attributes are represented by dummy 0/1 variables. Speech quality which has 3 possible values is represented by three dummy binary variables.

Because the attributes under investigation are clearly correlated, we use *ordinary least squares* (OLS) estimation for *multiple regression*. In order to apply OLS we remove all redundant dummy variables, namely the indicator of clean-speech, and normalize all variables to have zero mean. We calculate the partial regression coefficients by computing $b = (x'x)^{-1}x'y$, where *x* is the matrix of the values of the independent variables (attributes), *y* is the vector of the values of the dependent variable (CER), and *b* is the vector of partial regression coefficient.

Table 6. Ordinary least squares for multiple regression (with respect to CER). Dummy variables are listed in decreasing order of importance.

Attribute	Description : (Value)		Regression Coeff.
Style	Planned:(0)	Spontaneous:(1)	13.1
Network	CCTV4:(0)	Dragon:(1)	7.3
Gender	Female:(0)	Male:(1)	3.2
Channel	Studio:(0)	Telephone:(1)	2.7
Speech + Noise	No:(0)	Yes:(1)	2.5
Speech + Music	No:(0)	Yes:(1)	1.9
Speech Rate		char/sec:(|*rate* - 5.5|)	3.4
Length		Length:(*length*)	0.002

We applied the estimated linear regression function for predicting the CER of speaker turns (5990 in the dataset) and for predicting CER of speakers (138 in the dataset). For speaker turns, the regression predictor eliminated 19% of the variance compared to elimination of 33% of the variance by an optimal predictor assigning every speaker turn to the mean CER of the speaker. For speaker CER prediction, 53% of the variance is eliminated by the regression predictor.

5.2 Discussion

CER is highly dependent on the attributes we investigated. Table 7 lists extreme cases for which the CER is very high (31.5%) or very low (5.1%). The most important attribute found is the style which accounts to a 13.1% (absolute) increase in CER for spontaneous speech.

Table 7. CER computed for test subsets using top 3 most important attributes. CER standard deviation (σ) is computed using *bootstrapping*.

Attributes	CER	σ	Predicted CER
Planned, CCTV4, Female	5.7	0.5	6.3
Spontaneous, Dragon, Male	31.5	3.2	30.1

The second most important attribute found is the broadcasting network which may be attributed to topical differences between networks. Gender is also found significant (3.2%). Speech over a telephone channel suffers from a degradation of 2.7% but more data is needed for a reliable estimate. Degraded speech (music, noise) suffers from a degradation of about 2%. The speech rate is also an important factor – a degradation of 6% in CER is observed for high-rate speech.

Finally, the amount of data per tested speaker is not found to be significant in the regression. However, this is mostly due to the assumption that CER is a linear function of length. When using a binary dummy variable for length (short vs. long) we observe that for speakers with test data shorter than 30 seconds, we get a regression coefficient of 7.3 (7.3% degradation for speakers shorter than 30sec, compared to speakers longer than 30sec). For a 60 seconds threshold, we find a small regression coefficient of 0.2. The degradation for speakers shorter than 30sec may be due to insufficient adaptation data or to some other unknown phenomena.

6 Conclusions

In this work, we consider the Mandarin broadcast speech transcription task in the context of the DARPA GALE project. A state-of-the-art Mandarin speech recognition system is presented and validated on both BN and BC data. Experiments demonstrate that the MPE-based discriminative training leads to significant reduction in CER for this task. We also describe in this paper a topic-adaptive language modeling technique, and successfully apply the technique in the broadcast transcription domain. Lastly, a comprehensive error analysis is carried out to help steer future research efforts.

References

1. S. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in speech transcriptions at IBM under the DARPA EARS program," *IEEE Transactions on Audio, Speech, and Language Processing*, accepted for publication.
2 G. Saon, M. Padmanabhan, R. Gopinath, and S. Chen, "Maximimum likelihood discriminant feature spaces," in *Proc. ICASSP'00*, vol. 2, pp. 1129-1132, June 2000.
3. S. Wegmann, D. McAllaster, J. Orloff, and B. Peskin, "Speaker normalization on conversational telephone speech," in *Proc ICASSP'96*, vol. 1, pp. 339-343, May 1996.
4. M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech & Language*, vol. 12, no. 2, pp. 75-98, April 1998.
5. D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. ICASSP'02*, vol. 1, pp. 105-108, May 2002.
6. D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: discriminatively trained features for speech recognition," in *Proc. ICASSP'05*, vol. 1, pp. 961-964, March 2005.
7. S. F. Chen and J. T. Goodman, "An empirical study of smoothing techniques for language modeling," *Technical Report TR-10-98*, Computer Science Group, Harvard University, 1998.

8. K. Seymore and R. Rosenfeld, “Using story topics for language model adaptation,” in *Proc. Eurospeech '97*, September 1997.
9. L. Chen, L. Lamel, and J. L. Gauvain, “Lightly supervised acoustic model training using consensus networks,” in *Proc. ICASSP’04*, vol. 1, pp. 189-192, May 2004.
10. H. Y. Chan and P. C. Woodland, “Improving broadcast news transcription by lightly supervised discriminative training,” in *Proc. ICASSP’04*, vol. 1, pp. 737-740, May 2004.
11. M. J. F. Gales, A. Liu, K. C. Sim, P. C. Woodland, and K. Yu, “A Mandarin STT system with dual Mandarin-English output,” presented at GALE PI Meeting, Boston, March 2006.
12. B. Xiang, L.Nguyen, X. Guo, and D. Xu, “The BBN mandarin broadcast news transcription system,” in *Proc. Interspeech’05*, pp. 1649-1652, September 2005.
13. R. Sinha, M. J. F. Gales, D. Y. Kim, X. A. Liu, K. C. Sim, P. C. Woodland, “The CU-HTK Mandarin broadcast news transcription system,” in *Proc. ICASSP’06*, May 2006.

Improved Large Vocabulary Continuous Chinese Speech Recognition by Character-Based Consensus Networks

Yi-Sheng Fu, Yi-Cheng Pan, and Lin-shan Lee

Speech Lab, College of Electrical Engineering and Computer Science, National Taiwan University
mayaplus@speech.ee.ntu.edu.tw, thomas@speech.ee.ntu.edu.tw,
lslee@gate.sinica.edu.tw.

Abstract. Word-based consensus networks have been verified to be very useful in minimizing word error rates (WER) for large vocabulary continuous speech recognition for western languages. By considering the special structure of Chinese language, this paper points out that character-based rather then word-based consensus networks should work better for Chinese language. This was verified by extensive experimental results also reported in the paper.

1 Introduction

Substantial efforts have been made by many researchers to try to optimize the large vocabulary continuous speech recognition (LVCSR) process via different approaches. In the conventional maximum a posteriori probability (MAP) decoding [1] the recognizer selects the word hypothesis string with the highest string posterior probability given the acoustic and language model scores. Following such an approach, in principle the expected sentence error rate is minimized. However, in speech recognition usually it is the minimized word error rate (WER) rather than the minimized sentence error rate which is desired. A different approach was therefore developed to explicitly minimize the WER in an N-best rescoring algorithm [2]. However, this approach only solves the problem in a suboptimal way, because constrained by the N-best lists the hypothesis space is reduced to a rather small subset compared to the search space of the original recognizer. A new WER minimization algorithm applicable to word lattices was then developed [3]. Since word lattice is a very compact intermediate representation of alternative hypotheses and includes orders of magnitude more hypotheses than the typical N-best lists, searching through the entire word lattices usually requires too much computations. As a result, in the above approach [3] the word lattice is first reduced to a word-based consensus network as shown in Figure 1, which is the cascade of several properly aligned segments, each of which includes several word hypotheses having consensus on their word entities (the same word or words with similar phonemic structure) and time spans (reasonably overlapping in time), such as $w_1, w_1^{'}, w_1^{''}$ in Figure 1. By choosing the word hypotheses having the highest posterior probabilities in each segment, the final word string can be obtained with the expected WER minimized to a good extent [3].

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 422 – 434, 2006.

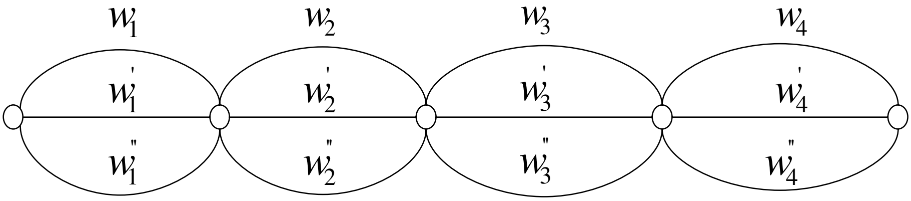


Fig. 1. A word-based consensus network to minimize WER in a LVCSR system

Although the above approach of word-based consensus network [3] was very successful for western languages, in this paper we point out that it is not well suitable for Chinese language due to the mono-syllable/character structure of Chinese language. As a result, a character–based consensus network considering the mono-syllable/character structure of Chinese language is proposed in this paper, and extensive experimental results indicated that with this character-based consensus network improved recognition performance can be obtained as compared to the word-based consensus network, or the conventional one-pass or two-pass search algorithms. Similar concept of evaluating the character posterior probabilities has been independently developed and proposed by different considerations with improved performance demonstrated [4, 5, 6], but here we look at the problem from a different point of view, and verify the concept with extensive experiment results.

Below, we will first review the basic principles from MAP decoding to the word-based consensus network in section 2. The mono-syllable/character structure of Chinese language and its impact on word-based consensus networks are then discussed in section 3. The new character-based consensus network proposed here in this paper is then presented in section 4, with experimental results reported in section 5.

2 Word-Based Consensus Network

Here the basic principle from MAP decoding to word-based consensus network is first briefly reviewed and carefully formulated such that the basic principles of character-based consensus network can be easily obtained from the formulation here.

2.1 From MAP Decoding to Minimum WER with N-Best Rescoring

Given a sequence of acoustic feature vectors A, suppose the probability distribution of each possible word string R, $P(R \mid A)$ is available, then the expected loss of decoding the input A into a specific output word string $W = w_1 w_2 w_3 .....w_N$ based on the distribution $P(R \mid A)$, $E_{P(R|A)}[l(W,R)]$, can be given by

$$E_{P(R|A)}[l(W,R)] = \sum_R P(R \mid A) l(W,R), \qquad (1)$$

where $l(W,R)$ is the loss function for a given pair of W and R. The goal of decoding is then to find the word string W that minimizes $E_{P(R|A)}[l(W,R)]$ as given above.

When we define the loss function $l(W,R)$ as the sentence error $SE(W,R)$,

$$l(W,R) = SE(W,R) = \begin{cases} 1 & if \quad W \neq R \\ 0 & if \quad W = R \end{cases}$$

equation (1) is reduced to

$$E_{p(R|A)}[l(W,R)] = \sum_{\{R|R\neq W\}} P(R \mid A) = 1 - P(W \mid A), \tag{2}$$

with which the minimum expected loss is equivalent to the maximum posterior probability $P(W \mid A)$, or the MAP principle. Therefore with the MAP principle the expected sentence error rate is minimized.

However, for most recognition system it is the minimized word error rate (WER) rather than the minimized sentence error which is desired. In that case, we can reformulate the loss function $l(W,R)$ as the word error $WE(W,R)$, or the number of different words between W and R, and we then have

$$E_{P(R|A)}[l(W,R)] = \sum_{R} P(R \mid A) WE(W,R), \tag{3}$$

If the word strings R in the above equation (3) are reduced to the N-best list, the above equation becomes

$$E_{P(R|A)}[l(W,R)] = \sum_{i=1}^{N} P(R_i \mid A) WE(W,R_i), \tag{4}$$

where R_i is the i-th word string in the N-best list. The goal of decoding is then to find the string W among the N-best lists $\{R_i\}$ which minimizes the expected loss in equation (4). This approach is known as the WER minimization by an N-best rescoring algorithm [2].

2.2 Word-Based Consensus Network for WER Minimization

Word lattices represent a combinatorial number of sentence hypotheses, offering very good potential to improve the above N-best rescoring approach through both an accurate word error estimates [$WE(W,R)$ in equation (3)] and a much larger search space for minimization than that in (4). The lattice actually leads to a computational problem. The number of word string hypotheses represented in a lattice is usually several orders of magnitude larger than that in an N-best list of reasonable size, which makes the straightforward computation of equation (4) infeasible.

The word-based consensus network algorithm is an engineering solution to the above computational problem. In this approach a segmentation process is first developed to divide the original lattice into K segments, or to divide each path in the lattice into K segments, each including a word hypothesis. After the segmentation, each segment includes several word hypotheses from different paths in the lattice but having consensus on their word entities and time spans, such as $w_1, w_1^{'}$, and $w_1^{''}$ in Figure 1 in the above. The cascade of these segments is thus a word-based consensus network. Equation (3) can be evaluated by

$$\begin{aligned}\sum_R P(R \mid A) WE(W,R) &= \sum_R P(R \mid A)\left[\sum_{k=1}^{K} WE(w_k, r_k)\right] \\ &= \sum_{k=1}^{K}\sum_R P(R \mid A) WE(w_k, r_k)\end{aligned} , \quad (5)$$

where w_k, r_k are the word hypotheses in the k-th segment for the word strings W and R, respectively. We can now redefine the word error as

$$\begin{cases} WE(w_k, r_k) = 1 & if \quad w_k \neq r_k \\ WE(w_k, r_k) = 0 & if \quad w_k = r_k \end{cases}$$

Equation (5) can then be simplified as follows:

$$\begin{aligned}\sum_{k=1}^{K}\sum_R P(R \mid A) WE(w_k, r_k) &= \sum_{k=1}^{K} P\left(\left\{r_k \mid r_k \in R, r_k \neq w_k\right\} \mid A\right) \\ &= \sum_{k=1}^{K}\left[1 - P(w_k \mid A)\right]\end{aligned} , \quad (6)$$

In this way the difficult problem of minimizing equation (3) becomes computationally tractable now. The optimal word string W minimizing equation (6) is apparently the cascade of those word hypothesis having the maximal word posterior probabilities $P(w_k \mid A)$ in each segment k, $k = 1......K$.

3 The Problem of Word-Based Consensus Network for Chinese Language

Chinese language is quite different from many western languages such as English. It is not alphabetic. Chinese characters are idiographic symbols. Almost each Chinese character is a morpheme with its own meaning. A word is composed of one to several characters, with meaning somehow related to the meaning of the component characters. A nice property is that all the characters are pronounced as monosyllables, and the total number of phonologically allowed monosyllables is limited, roughly 1345 for Mandarin. But the number of commonly used characters is much higher, roughly 8000 to 10000. This implies the existence of large number of homonym characters sharing the same monosyllable. As a result, each monosyllable represents many characters with different meanings, and the combination of these monosyllables

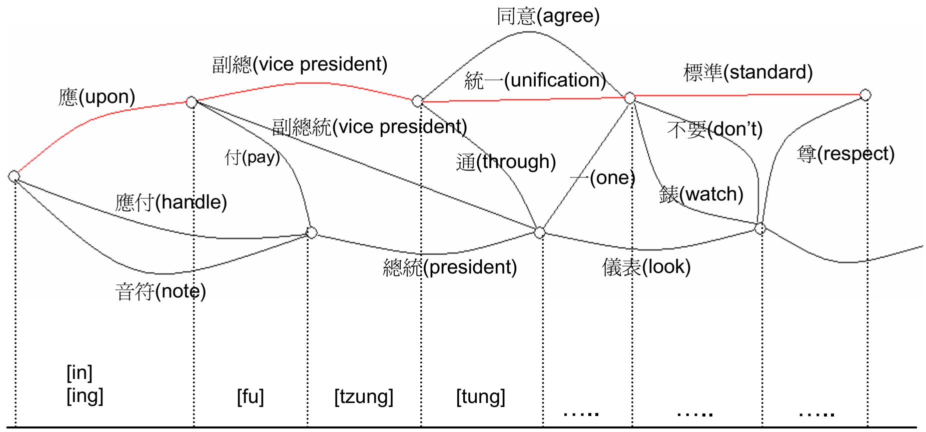


Fig. 2. The difficulties in generating good word-based consensus network for Chinese language

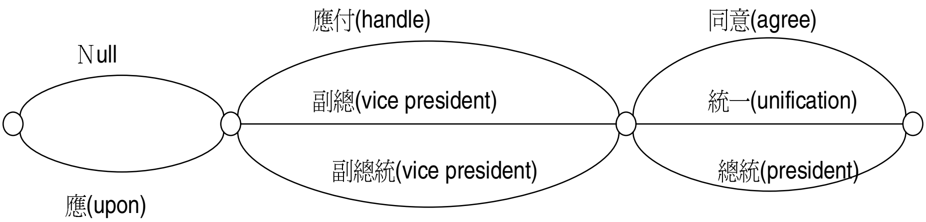


Fig. 3. Word-based consensus network directly obtained from the first several word arcs of the word lattice in Figure 2

(or characters) gives almost unlimited number of words (at least 80000 are commonly used) and sentences. This is referred to as the mono-syllable/character structure of Chinese language here.

The above mono-syllable/character structure of Chinese language makes the previously mentioned word-based consensus network difficult to use. Consider an example utterance, "應副總統一標準(的指示)…(Upon (the instruction of) unifying the standard by the vice president…)", a partial list of the possible word lattice generated is shown in Figure 2. The correct path is represented by red arcs. The first syllable [ing] is for the first monosyllabic word "應(upon)", while the next two syllables [fu] and [tzung] form the second bisyllabic word "副總(vice president)", etc. However, the first two syllables can also form other two noisy word hypothesis "應付(handle)" and "音符(note)", while the third and fourth syllables [tzung] and [tung] can form another noisy word hypotheses "總統(president)" and so on. The majority of Chinese words are bisyllable. From Figure 2, it can be easily found that very possibly a syllable hypothesis may connect to the syllable hypotheses on its left or right to form different word hypotheses. Correct identification of word boundaries from an utterance is usually difficult too. The segmentation algorithm developed for constructing word-based consensus networks for English can certainly work here, but with much more errors. Take the example in Figure 2, the monosyllabic word hypothesis “應(upon)”, the bisyllabic word hypotheses “應付(handle)”, “副總(vice president)”, “總統 (president)” and the trisyllabic word hypothesis “副總統(vice president)” may all be highly probable word arcs, but it is difficult to construct a good word-based consensus network here, because there are high degree of ambiguities among the word entities and time spans of the word hypotheses, or it is difficult to locate the boundaries of the segments for a word based consensus network. Following the standard segmentation algorithm for English, the word-based consensus network obtained for the first several word arcs in Figure 2 may look like that in Figure 3, which may result in serious recognition errors. For example, the first mono-character word “應(upon)” may be deleted and replaced by a “Null element”, and there may be additional errors for the next several segments as well.

4 Character-Based Consensus Network for CER Minimization

With the above considerations, here in this paper we propose a similar but slightly different approach to tackle the problem: using the sub-word unit of character as the unit in the segmentation process to construct character-based consensus networks rather than word-based consensus networks. It is easy to see from Figure2 that although the consensus among word arcs are very limited and it is difficult to construct good word-based consensus networks, the syllable or character boundaries have much higher consensus, and the consensus in character entities and time spans are also much higher given a word lattice. Generally speaking, it is very easy to record transition time between each HMM states. Therefore the time information of each character of a word is handy after decoding.

In addition, each character in Chinese has its own meaning and plays linguistic roles rather independently in sentences. Also, there are no “blanks” in written or printed Chinese sentences serving as word boundaries as in western languages. As a result, the “words” in Chinese have very flexible structure and are not very well defined, the segmentation of a sentence into words is usually not unique, and there even does not exist a commonly acceptable lexicon of words. This is why in Chinese speech recognition consistent evaluation of WER is difficult, and the performance evaluation is usually based on character error rates (CER). So it is also reasonable to perform the recognition over the character-based consensus network proposed here to try to minimize CER.

Note that similar concept of evaluating the character posterior probabilities has been independently developed [4, 5, 6], but here we propose the character-based consensus network from a different point of view, and later on in the paper extensive experimental results will be presented to verify the concept here.

4.1 Character Lattices and Character Posterior Probabilities

The character-based consensus network is constructed based on a character lattice transformed from the word lattice and the character posterior probabilities. The character posterior probabilities can be obtained directly from the popularly used formula for word posterior probabilities given a word lattice [7]. Given an acoustic signal A the word posterior probability for a specific word arc in the obtained word lattice with the respective begin-time τ and end-time t can be estimated by the following equation (7):

$$P([w;\tau,t]\,|\,A)=\frac{\sum\limits_{\substack{path\ W\in Lattice:\\ W\ contains\ [w;\tau,t]}} P(W\,|\,A)}{\sum\limits_{path\ W'\in Lattice} P(W'\,|\,A)}=\frac{\sum\limits_{\substack{path\ W\in Lattice:\\ W\ contains\ [w;\tau,t]}} P(W)\cdot P(A\,|\,W)^{1/\lambda}}{\sum\limits_{path\ W'\in Lattice} P(W')\cdot P(A\,|\,W')^{1/\lambda}},\ (7)$$

Where the word lattice is used as the space of all word string hypotheses considered and the summations are over all paths in the lattice and all paths in the lattice but contains $[w,\tau_c,\tau_t]$ in the denominator and numerator respectively. And $P(W)$ represents the probability obtained with the language model. For character lattice and character posterior probability, we can easily transform the word lattice into a corresponding character lattice by simply dividing each word arc in the word lattice into its component character arcs as in Figure 4. The posterior probability for each component character arc is then simply the posterior probability of the original word arc, because all the paths in the character lattice and containing the considered component character arc are exactly the same as those for the original word arc. A typical character lattice obtained in this way from the first several word arcs in Figure 2 is shown in Figure 4. Note that the structure of the character lattice in Figure 4 is exactly the same as that of the original word lattice in Figure 2. So only those component character arcs of the same word arc in the original word lattice can be connected here. In this way much of the ambiguities can be avoided.

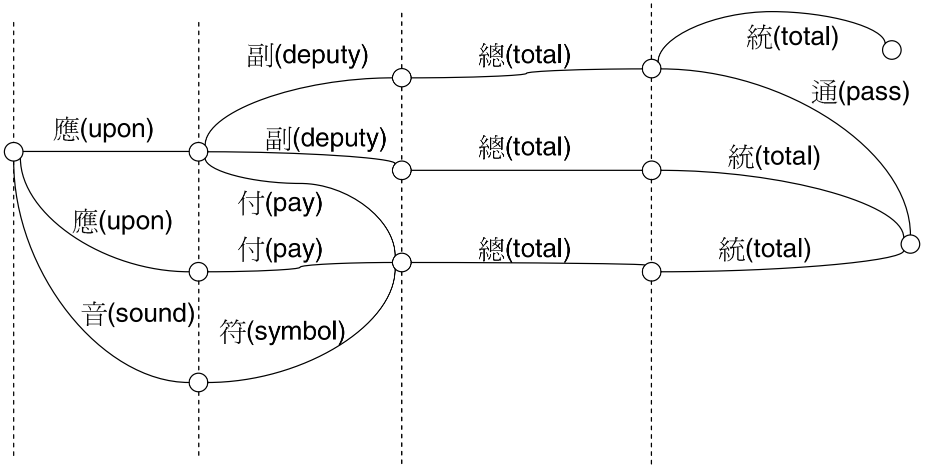


Fig. 4. Corresponding character lattice transformed from the first several word arcs in the word lattice in Figure 2

4.2 Character-Based Consensus Network

With the character lattice and corresponding character posterior probabilities obtained above, we can then construct the character-based consensus network by a clustering process very similar to that of constructing the word-based consensus network. First we regard each character arc in the character lattice as a separate cluster. We may first need to prune those character arcs with extremely low posterior probabilities, followed by the clustering approach with details given below, in which we merge some clusters together to form bigger clusters, and the process continues recursively. After this clustering process is finished, the final sets of clusters are sorted according to their time indices, and this produces the final character-based consensus network. The clustering approach mentioned above includes two basic steps, intra-character first and inter-character next, both performed recursively.

In the first step of intra-character clustering, the goal is to merge those character arcs carrying exactly the same character and have good consensus in time spans, such as the two character arcs "應(upon)" in the beginning of the character lattice in Figure 4. For any two clusters C_1 and C_2 carrying the same character, we may define its similarity $SIM_1(C_1,C_2)$ as

$$SIM_1(C_1,C_2)=\max_{\substack{c_1\in C_1\\ c_2\in C_2}} overlap(c_1,c_2)\cdot p(c_1)\cdot p(c_2)\ , \tag{8}$$

where c_1,c_2 are character arcs in the clusters C_1,C_2 respectively, $overlap(c_1,c_2)$ is the length of the overlapped time period between arcs c_1 and c_2 normalized by the sum of their respective lengths (i.e., the lattice alignment requirement), and $p(c_1), p(c_2)$ are the posterior probabilities for c_1,c_2. In the beginning each cluster C_i contains only one character arc and in each iteration we merge two clusters C_1 and C_2 that having the highest similarity $SIM_1(\mathrm{C}_1,\mathrm{C}_2)$. So those clusters with higher posterior probabilities and longer overlapping time period will be merged first. After merging, the posterior probabilities of the new cluster is simply the sum of the posterior probabilities of the two component clusters, because it is easy to see from equation (7) the denominators for the two posterior probabilities are the same, while the numerators are additive.

In the second step of inter-character clustering, the goal is to merge those character arcs carrying different characters but having good consensus in phonemic structures and time spans, such as the character arcs "應(upon)" and "音(sound)" in the beginning of the character lattice in Figure 4, in which case the right vertices of these two character arcs will be merged. Similarly as in equation (8), the similarity $SIM_2(\mathrm{C}_1,\mathrm{C}_2)$ for two clusters C_1 and C_2 can be defined as

$$SIM_2(\mathrm{C}_1,\mathrm{C}_2)=\underset{\substack{c_1\in \mathrm{C}_1\\ c_2\in \mathrm{C}_2}}{avg}\ \Lambda(c_1,c_2)\cdot p(c_1)\cdot p(c_2)\,, \tag{9}$$

Where C_1, C_2 are clusters carrying different characters, $\Lambda(c_1, c_2)$ is the phonemic similarity between the two component character arcs c_1 in C_1 and c_2 in C_2, and $p(c_1)$, $p(c_2)$ are the character posterior probabilities of c_1 and c_2. So the clusters C_1, C_2 with the highest similarity $SIM_2(\mathrm{C}_1, \mathrm{C}_2)$ will be merged first, and both ends of the two character arcs for the clusters C_1, C_2 will be merged, with each arc having its original posterior probabilities. In addition, for two clusters C_1, C_2 to be merged they have to satisfy the lattice alignment requirement, i.e. all component character arc c_1 in C_1 has to overlap with all component character c_2 in C_2 for the same time.

After the clustering process, all the clusters are sorted by their time information and the character-based consensus network is constructed. During decoding, in every segment of the character-based consensus network, the character with the highest posterior probability is chosen.

5 Experimental Results

Test results are reported here.

5.1 Test Corpus and Baseline System

The speech corpus we used is the broadcast news corpus collected from the local radio station in Taiwan in 2001. The baseline system is a one-pass tree-copy trigram decoder, and the baseline word lattice is obtained by a bigram decoding phase. In the generation of the word lattices we adjusted the word lattice beam width to have different number of word arcs retained in the bigram decoding phase. With the given lattices we also performed a second-pass rescoring using trigram language model to find the path with the maximal MAP scores. Initial/Final acoustic models and the feature vectors of 13 MFCC and their first and second-order time derivatives were used. The test set includes 87 news stories with total length of 40 minutes and a total of 11917 characters. It is noted here that in section 4.1 we've mentioned the posterior probability for each component character arc is then simply the posterior probability of the original word arc, therefore the trigram language model rescoring can also be conducted on the character-based consensus network.

5.2 Performance of Word-Based Consensus Network

The baseline one-pass trigram decoder results in terms of character accuracy (1-CER) for different word lattice beam width are listed in the second column of Table 1. We also compared the two-pass rescoring results listed in the next column, in which in the first phase we generated the same word lattices by bigram decoding to be used in the following word- or character-based consensus network approach, and then in the second phase we used trigram language model to rescore the word lattices. It can be found that the rescoring approach offer slightly worse performance than the one-pass

Table 1. Performance comparison of the various speech decoding approaches

Word lattice beam width	One-Pass Trigram decoder	Two-pass Rescoring	Word-based consensus network(WCN)		Character-based consensus network(CCN)	
			Language model weight	Character accuracy	Language model weight	Character accuracy
50	76.19	75.77	9	75.80	9	76.37
			10	75.99	10	76.49
			11	**76.25**	11	**76.73**
			12	76.22	12	76.60
			13	76.15	13	76.56
55	76.19	75.82	9	75.84	9	76.50
			10	76.10	10	76.68
			11	76.32	11	**76.89**
			12	**76.33**	12	76.74
			13	76.26	13	76.58
60	76.19	75.91	9	76.09	9	76.72
			10	76.37	10	76.96
			11	**76.58**	11	**77.11**
			12	76.54	12	77.02
			13	76.45	13	76.90
65	76.19	75.97	9	76.19	9	76.78
			10	76.44	10	77.01
			11	**76.65**	11	77.12
			12	76.64	12	**77.18**
			13	76.48	13	77.04
70	76.19	76.04	9	76.28	9	76.86
			10	76.52	10	77.07
			11	**76.73**	11	77.23
			12	76.66	12	**77.26**
			13	76.61	13	77.12

trigram decoder, which is a known fact. In the next column of the table we list the results for word-based consensus network under different setting of the language model weight in equation (7), we see we can consistently get better results using the word-based consensus network approach as compared with the two-pass rescoring approach with all choices of the language model weight. Furthermore, in many cases the performances of word-based consensus network can even be better than the baseline one-pass trigram decoding, if the language model weight was properly chosen. This is where minimizing the word error in each segment in equation (6) turned out to be better than minimizing the sentence error in equation (2). This verified the effectiveness of the word-based consensus network, even for evaluation based on CER.

5.3 Performance of Character-Based Consensus Network

The results for the character-based consensus network approach are in the last column of the table. We can see that under all different language model weight and all different word lattice beam width, the character-based consensus network provided consistently better results than the word-based consensus network approach. These results in Table 1 are summarized in Figure 5, where the results using the best choices of the language model weights are shown. In addition, the performance of the proposed character-based consensus network approach is actually always better than the word-based consensus network as well as the baseline one-pass trigram decoder approach under all parameter settings. It can also be found in Figure 5 that the performance of character-based consensus network was improved continuously as the word lattice bandwidth was increased. It seemed not yet saturated for the maximum bandwidth of 70 shown in Figure5. This verified the outstanding performance of the proposed character-based consensus network. The case for word lattice beam width being 70 with all choices of language model weights are also plotted in Figure 6 as an example.

We can then take a deeper look into the error types for the word- and character-based consensus network as shown in Table 2. We noticed that the number of substitution errors in the character-based consensus network is slightly higher than that in the word-based consensus network, while the number of deletion errors in character-based consensus network is significant lower. This phenomenon is consistent with our early discussions that it is difficult to construct good word-based consensus networks from word lattices for Chinese language. The high degree of ambiguities among the word entities and time spans of the word arcs in the word lattice causes major problem in assigning the word arcs into proper segments of the

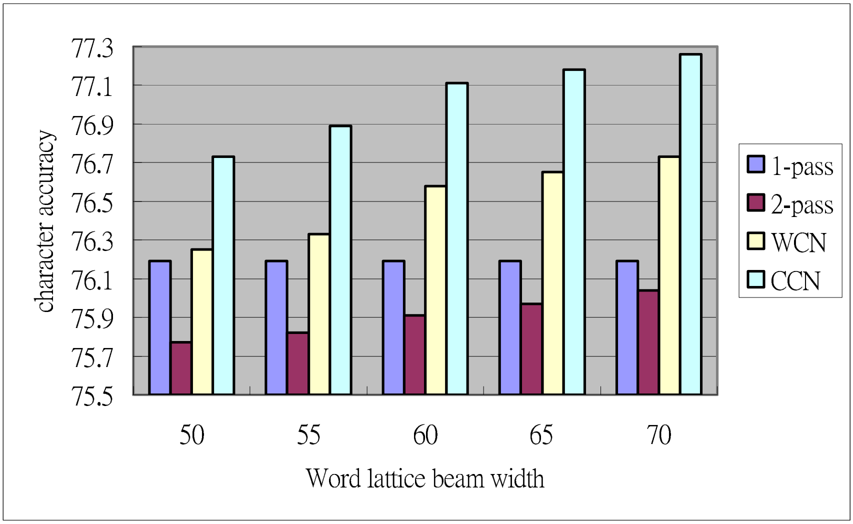


Fig. 5. Summary of Table 1: the comparison among the different decoding approach discussed here

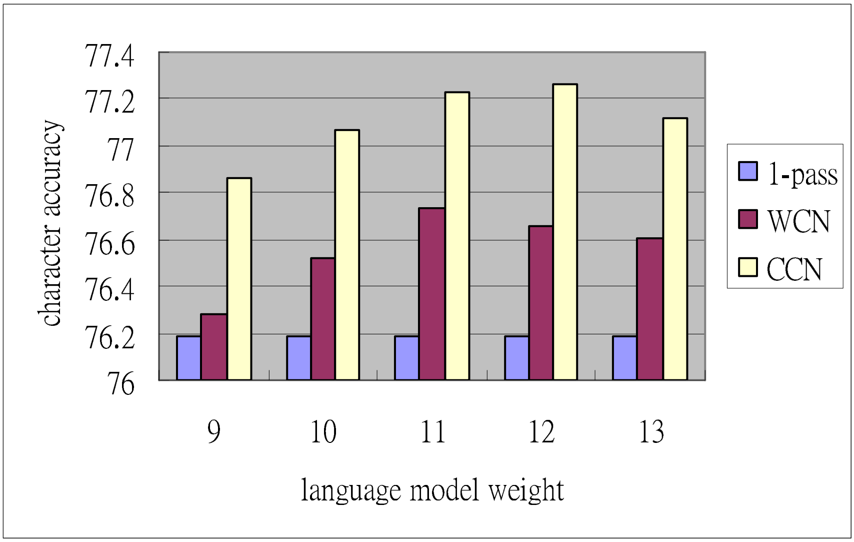


Fig. 6. Performance of word- and character-based consensus networks compared with 1-pass trigram decoder for word lattice beam width being 70 and different language model weights

Table 2. Error types in the two different consensus networks

Word lattice beam width	**Language model weight**	**Word-based consensus network**			**Character-based consensus network**		
		Insertions	Deletions	Substitutions	Insertions	Deletions	Substitutions
50	9	98	364	2422	91	228	2497
	10	97	365	2399	90	233	2479
	11	90	376	2364	92	245	2436
	12	85	372	2377	89	255	2444
	13	85	371	2386	88	259	2446
	14	87	382	2399	86	257	2468
	15	86	392	2390	86	259	2470
	16	85	391	2406	84	266	2485

word-based consensus network. Once the assignment is biased for some specific segment with higher posterior probabilities, the adjacent segment may then have too few arcs and result in a deletion error. One example is in Figure 3, in which the second segment is biased and a deletion is very possibly generated in the first segment. This is why the character–based consensus network gives much less deletion errors, which in turn leads to slightly higher substitution errors. In other words, with more deletions in the word-based consensus networks, some character hypotheses which may be misrecognized were deleted as well. So the substitution errors may become less.

6 Conclusion

In this paper, we first briefly review the concept of word-based consensus network, and discuss why it is not suitable for Chinese language. We then propose an approach to optimize the obtainable performance by character-based consensus network, which is more suitable for Chinese language. Extensive experimental results verified that the proposed character-based consensus network performed better than the word-based consensus network for Chinese language.

References

1. Bahl, L. R., Jelinek, F & Mercer, R.L. (1983). "A Maximum Likelihood Approach to Continuous Speech Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, 5(2), 197-190
2. Stolcke, A., Konig, Y. and Weintraub, M., "Explicit Word Error Minimization in N-best List Rescoring", Proc. Eurospeech, pp. 163-166, 1997.
3. Mangu, L., Brill, E. and Stolckes, A., "Finding Consensus in Speech Recognition: Word Error Minimizaiton and Other Applications of Confusion Networks", Computer Speech and Language, Vol.14, No.4, pp.373-400, 2000.
4. Soong F.K., Lo W. K. and Nakamura, S., "Generalized Word Posterior Probability (GWPP) for Measuring Reliability of Reconized Words", Proc. SWIM 2004.
5. Soong F.K., Lo W. K. and Nakamura, S., "Optimal Acoustic and Language Model Weights for Minimizing Word Verification Errors", Proc. ICSLP 2004.
6. Yao Qian, Soong F.K., Tan Lee, "Tone-enhanced Generalized Character Posterior Probability (GCPP) for Cantonese LVCSR" Proc. ICASSP 2006.
7. Wessel, F., Schluter, R. and Ney, H., "Using Posterior Probabilities for Improved Speech Recognition", Proc, ICASSP, 2000

All-Path Decoding Algorithm for Segmental Based Speech Recognition

Yun Tang, Wenju Liu, and Bo Xu

National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Sciences
hottang@gmail.com, lwj@nlpr.ia.ac.cn

Abstract. In conventional speech processing, researchers adopt a dividable assumption, that the speech utterance can be divided into non-overlapping feature sequences and each segment represents an acoustic event or a label. And the probability of a label sequence on an utterance approximates to the probability of the best utterance segmentation for this label sequence. But in the real case, feature sequences of acoustic events may be overlapped partially, especially for the neighboring phonemes within a syllable. And the best segmentation approximation even reinforces the distortion by the dividable assumption. In this paper, we propose an all-path decoding algorithm, which can fuse the information obtained by different segmentations (or paths) without paying obvious computation load, so the weakness of the dividable assumption could be alleviated. Our experiments show, the new decoding algorithm can improve the system performance effectively in tasks with heavy insertion and deletion errors.

1 Introduction

In the framework of statistics, the mapping between the label sequence $l_1^N = \{l_1, l_2, \cdots, l_N\}$ and one speech observation sequence $x_1^T$is determined by the rule of maximum a posterior probability (MAP).

$$\begin{aligned} l_1^N &= \arg\max_{l_1^N, N} P(l_1^N | x_1^T), \\ &= \arg\max_{l_1^N, N} P(x_1^T | l_1^N) P(l_1^N), \end{aligned} \tag{1}$$

where, $P(x_1^T|l_1^N)$ is the acoustic model score and $P(l_1^N)$ is the language model score for the label sequence l_1^N. Each label in l_1^N represents an acoustic event, such as phonme, and is exhibited by a corresponding segment in x_1^T. The corresponding segments for neighboring labels may be disjunct, or adjacent, or even overlapped. For example, the corresponding segments for label sequence $\{a, b, c, d, e\}$ are shown in fig.1(I), where "sil" means that there exists silence between labels. Shadowed parts in fig.1(I) indicate that these regions in time axis are shared by more than one acoustic event. The corresponding observation segments for a and b are disjunct; the corresponding observation segments for b

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 435–444, 2006.

and c are overlapped partially. For the convenience of modeling and decoding, a dividable assumption is adopted in the state-of-the-art of acoustic model, that is, the corresponding segment for each label is dividable in time axis and no overlapping region exists, as fig.1(II) shows. By this assumption, the influence of each acoustic event is limited in its corresponding observation segment, such as the context independent acoustic model; or it may also affect neighboring observation segments, such as the context dependent acoustic model. Then, the likelihood probability of a label sequence matching an observation sequence could be dissolved into likelihood scores of labels on their corresponding segments.

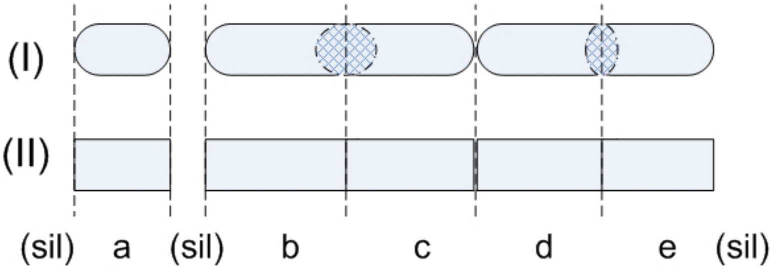


Fig. 1. Dividable assumption in speech processing. (I) The overlapping between acoustic events; (II) the segmentation after the dividable assumption is applied.

$$P(x_1^T|l_1^N) = \sum_{B_0^N \in \Lambda_{T,N}} \prod_{i=1}^{N} P(x_{B_{(i-1)}+1}^{B_i}|l_i), \tag{2}$$
$$B_{i-1} < B_i, B_0 = 0, B_N = T,$$

where $\Lambda_{T,N}$ is the segmentation set for an observation sequence with T frames and N labels, $(B_{i-1} + 1)$ and B_i are the boundary frames of the i-th label. As N and T increasing, the number of possible segmentations is increased exponentially. So it is infeasible to count all segmentation cases to find the most likely label sequence for a given observation sequence. An usual way is to choose the label sequence with the highest probability segmentation as an approximate result of Equ.(2)

$$P(x_1^T|l_1^N) = \max_{B_0^N \in \Lambda_{T,N}} \prod_{i=1}^{N} P(x_{B_{i-1}+1}^{B_i}|l_i). \tag{3}$$

As mentioned above, the boundary between neighboring labels may be overlapped or even not exist for certain label pairs, such as the virtual syllable initial and syllable final pairs for non-initial syllables[1] in Mandarin[1]. Hence, the dividable assumption does not represent the real situation of human utterances. The

[1] The syllables only consist of syllable final and tone in Chinese characters, such as digit ”2”.

best path approximation in Equ.(3) ignores other possible segmentations and reinforces the distortion introduced by the dividable assumption. In this paper, we propose an all-path decoding algorithm, which can fuse the information obtained by different segmentations or paths without paying obvious computation load, so the weakness of the dividable assumption could be alleviated. In order to distinguish the all-path decoding from the conventional method, the decoding based on Equ.(3) is called the best path decoding in this paper. Our experiments show, the best path approximation has no obvious influence on common tasks, while the all-path decoding can effectively improve the system performance in tasks with heavy reduction and deletion errors.

The rest of this paper is organized as follows. A brief introduction of Stochastic Segment Model(SSM)[2], the acoustic model adopted in the prototype system, is given in Section 2. Then in Section 3, the all-path decoding algorithm is presented in detail. Section 4 shows the experimental results and analysis on the mandarin digit string and mandarin large vocabulary continue speech recognition (LVCSR) tasks respectively. Conclusions are drawn in the last section.

2 Stochastic Segment Model

Segment model (SM)[3] is a family of methods that adopt segmental distribution rather than frame-based features to represent the underlying trajectory of the observation sequence. The biggest difference between HMM and SM is that HMM models and decodes the utterance in a frame-based way while SM is in a segment-based way. A segment in SM can present a phoneme, a syllable or a word etc. The decoding algorithm of SM is directly based on Equ.(3).

SSM is one kind of SM. Each segment model in SSM has a fixed length region sequence r_1^L, which is used to represent the variable length observation sequence $x_{\tau_1}^{\tau_2}$. A re-sample function is needed to uniform $x_{\tau_1}^{\tau_2}$ to an L length sequence y_1^L so it can be measured by SSM.

$$y_i = x_{\lfloor \frac{i}{L}(\tau_2-\tau_1)+\tau_1 \rfloor}, 0 < i \leq L, \tag{4}$$

where $\lfloor z \rfloor$ is the maximum integer no larger than z.

The re-sampled frame is measured by region, which is similar to the conception of the state in HMM. The log-likelihood of a segment $x_{\tau_1}^{\tau_2}$ given model l_α is the production of region scores:

$$\ln[p(x_{\tau_1}^{\tau_2}|l_\alpha)] = \sum_{i=1}^{L} \ln[p(y_i|l_\alpha, r_i)] \tag{5}$$

where r_i is the i-th region model in segment model l_α. Usually, each region is characterized by a Gaussian mixture model.

From Equ.(3), the decoding process for utterance $x_1^T$is as follows:

$$J^*(m) = \max_{\tau, l_\alpha}\{J^*(\tau) + \ln[p(x_\tau^m|l_\alpha)] \cdot (m-\tau) + C\}, \tag{6}$$

where C is the insertion factor, $J^*(m)$ is the accumulated score of the best acoustic model sequence at frame m and $J^*(0)$ is initialized to 0. A candidate set and an expanding set are formed at each frame during decoding. The candidate set is a collection of hypothesized paths ending at this point and the expanding set is the collection of acoustic models which succeed the paths in the candidate set. The decoding is performed from 1 to T frame by frame and the decoding result is the label sequence attached to the path with the highest probability in candidate set of T.

3 All-Path Decoding Algorithm

The all-path decoding algorithm aims to find the probability of a label sequence for an utterance by integrating more information from passible segmentations or paths as Equ.(2) does. In the conventional best path decoding algorithm, a label sequence can reach frame m by multiple paths, which will be merged and only the path with the highest score is survived to the following decoding process. While, in the all-path decoding algorithm, these paths will be fused to form a comprehensive score. Assuming frame τ is ahead of frame m and we have obtained the probability of label sequence $l_1^n = \{l_1, l_2, \cdots, l_n\}$ reaching frame τ through all paths. We call the probability $P_A(x_1^\tau | l_1^n)$ as the all-path probability for l_1^n on x_1^τ. Then, at condition that the segment x_τ^m is labeled as l_{n+1}, the probability of label sequence l_1^{n+1} on x_1^m is,

$$P_A(x_1^m | \tau, l_1^{n+1}) = P_A(x_1^\tau | l_1^n) \cdot P_A(x_\tau^m | l_{n+1}) \cdot P(l_{n+1} | l_1^n). \tag{7}$$

Hence, the all-path probability for labels l_1^{n+1} on observation sequence x_1^m can be obtained by adding all $P_A(x_1^m | \tau, l_1^{n+1})$s with different τs ahead of m, that is,

$$P_A(x_1^m | l_1^{n+1}) = \sum_{\tau=1}^{m-1} P_A(x_1^m | \tau, l_1^{n+1}). \tag{8}$$

Consequently, the decoding process of SSM is modified according to Equ.(8) as follows,

$$J_A^*(m | l_1^{n+1}) = \{\ln \sum_{\tau=1}^{m-1} \exp[(f(m | \tau, l_1^{n+1}))/\beta]\} \cdot \beta, \tag{9}$$

$$f(m | \tau, l_1^{n+1}) = J_A^*(\tau | l_1^n) + (m - \tau) \ln p(x_\tau^m | l_{n+1}) + C, \tag{10}$$

where β is a transfer factor for likelihood probability to probability, $J_A^*(m | l_1^{n+1})$ is the all-path version of $J^*(m | l_1^{n+1})$, and $f(m | \tau, l_1^{n+1})$ is the likelihood version of $P_A(x_1^m | \tau, l_1^{n+1})$. The probability differences among paths will be enhanced when the transfer factor is small, whereas the sub-optimal paths will play more

```
$m = 0, J_0^* = 0$;
while $m \leq T$ do
    $m = m + 1$
    foreach $l_1^{n+1} \in$ active expanding sets
        $\tau = \max\{m - L_{ext}, 0\}, J_A^{\tau-1}(m|l_1^{n+1}) = P_{Min}$
        while $\tau < m$ do
            measuring $f(m|\tau, l_1^{n+1})$,
            updating $J_A^{\tau}(m|l_1^{n+1})$ from $J_A^{\tau-1}(m|l_1^{n+1})$ by Equ.(11)
            $\tau = (\tau + 1)$
        repeat
        $J_A^*(m|l_1^{n+1}) = J_A^{m-1}(m|l_1^{n+1})$
        pruning low-likelihood label sequences ended in $m$
    repeat
repeat
get the best label sequence from the last utterance frame.    #
```

Fig. 2. All-path decoding algorithm

important roles in the all-path probability score. When the value of β is close to 0, the all-path probability will approximate to the maximum likelihood score in current paths. So the all-path decoding algorithm will at least achieve a comparable performance with the best path based decoding algorithm. The optimal β is chosen by experience in the experiments.

The all-path probability in Equ.(9) is updated in a sequential way. Assuming that the decoder moves to frame m and the current label sequence is l_1^{n+1}. The initial value of $J_A^*(m|l_1^{n+1})$ sets to a minimal score P_{Min}; then we count the log likelihood score $f(m|\tau, l_1^{n+1})$ for each frame τ before m, and these scores update $J_A^*(m|l_1^{n+1})$ in a sequential way by the difference between $f(m|\tau, l_1^{n+1})$ and $J_A^*(m|l_1^{n+1})$,

$$\begin{aligned} J_A^{\tau}(m|l_1^{n+1}) = \ln[&\exp((J_A^{\tau-1}(m|l_1^{n+1}) - f_{max})/\beta) \\ &+ \exp((f(m|\tau, l_1^{n+1}) - f_{max})/\beta)] \cdot \beta + f_{max}, \end{aligned} \tag{11}$$

$$f_{max} = \max\{J_A^{\tau-1}(m|l_1^{n+1}), f(m|\tau, l_1^{n+1})\}, \tag{12}$$

where $J_A^{\tau-1}(m|l_1^{n+1})$ and $J_A^{\tau}(m|l_1^{n+1})$ are all-path probabilities before and after updated by the score $f(m|\tau, l_1^{n+1})$. In this way, the range of β could be expanded to a larger scale without consideration of overflowing or underflowing errors. The details of the all-path decoding algorithm is listed in fig.2. L_{ext} is the allowed maximum segment duration.

Pruning technologies[4][5][6] are applied in the all-path decoding algorithm, since most paths pruned are wrong paths and their probability scores are close to 0. So it will not make obvious difference whether these wrong paths are considered.

In practice, we assume that two label sequences are same if the last two words are identical. In fact, after pruning, the paths in the expanding set of each frame are originated from a few label sequences and these paths are different from each other mostly by segmentations or the last words. So Equ.(9) can be simplified to

$$J^*_A(m|W_I, W_{II}) = \{\ln \sum_{\tau=1}^{m-1} \exp[J^*_A(\tau|W_I) + \ln(p(x^m_\tau|W_{II}) + C)\beta]\} \cdot \beta, \quad (13)$$

where W_{II} is the last word and W_I is the word previous W_{II} in the current label sequence respectively, $J^*_A(\tau|W_I)$ is the maximum all-path probability for the label sequence with the last word W_I on frame τ, and $J^*_A(m|W_I, W_{II})$ is the all-path score on frame m for the label sequence with W_{II} and W_I as the last two words. Such simplification can seamlessly integrate to a system with a bigram language model. When the trigram model is used, we can trace back to compare the last three words in the current label sequence and a more accurate result can be expected.

4 Experiments and Analysis

4.1 Mandarin Digit String Recognition

The all-path decoding algorithm was first verified by the mandarin digit string recognition task. Digit string recognition has achieved a satisfied performance in English [7]. However, due to the serious confusion among mandarin digits, the state-of-the-art of mandarin digital string recognition systems does not match that of the English counterpart. Mandarin is a monosyllabic and tonal language, in which a syllable is composed of a syllable initial, syllable final, and tone. Insertion or deletion errors mainly exist in non-syllable initial characters, e.g., "1," "2," and "5." If a digit's syllable final is similar to that of the non-syllable initial character followed immediately, it is difficult to segment the non-syllable initial character accurately and segmentation errors tend to occur, such as the confusability between "5" and "55." Substitution errors mainly occur among "6," "9," and "yiao" ("yiao" is the variation of "1"), or between "2" and "8," because of the similarity of their syllable finals. Insertion and deletion errors are high related with the accuracy of segmentation. Hence, mandarin digit string is a good platform for verifying the all-path decoding algorithm.

The data corpus used in this experiment was the digit string database built by Institute of Automation, CAS [8]. The database was collected from 55 male speakers (80 utterances per speaker). The speech of the first 40 speakers (ordered by the name of speakers) were taken as the training set and the data from the remaining 15 speakers as the test set in digit string experiments.

The baseline system[5] was based on the whole-word model and each SSM was sequentially composed of 40 regions and each region was modeled by 12 Gaussian mixtures. Acoustic features were 12 dimensions MFCC plus 1 dimension normalized energy and their 1st and 2nd order derivatives. The comparison of mandarin digit string recognition results of two systems, the all-path decoding based system and the best path decoding based system, was listed in table1. "*S.Corr*," "*Sub*," "*Del*," "*Ins*" and "*Err*" were the string correct, substitution, deletion insertion and word error rate respectively. Compared with the baseline, the all-path method reduced 14.0% relative word error rate and the total error

Table 1. Mandarin digit string recognition results achieved by the all-path decoding based system and best path based system

Decoder	S.Corr%	Sub%	Del%	Ins%	Err%
Best Path	95.00	1.02	0.27	0.35	1.64
All Paths	95.67	0.91	0.23	0.27	1.41

Table 2. Recognition results in Test-863 achieved by the all-path decoding based system and best path based system

Decoder	Sub%	Del%	Ins%	CER%
Best Path	12.9	0.1	0.0	13.0
All-Path	12.4	0.1	0.1	12.6

made by insertion and deletion was obviously decreased by 19.4%. It is useful to consider information from all paths instead of the best path in tasks with heavy insertion and deletion errors.

4.2 Mandarin LVCSR

The all-path decoding algorithm had also been run in a mandarin LVCSR system to test the algorithm performance on common tasks. The data corpus applied in LVCSR experiments was provided by Chinese National Hi-Tech Project 863 for Mandarin LVCSR system development [9]. 83 male speakers' data were employed for training (48373 sentences, 55.6 hours)and 6 male speakers' for test (240 sentences, 17.1 minutes). The acoustic feature adopted the same configures used in section 4.1.

The baseline system was a context-dependent triphone SSM system [4]. The search paths were organized by the lexical tree and began/ended with the silence model. Each segment model was sequentially composed of 15 regions and each region was modeled by 12 Gaussian mixtures. Region models were tied by phone based decision trees. Triphone based duration models were used to improve the system accuracy.

Table2 showed the recognition results on Test-863 by the best path decoding and all-path decoding. The "CER" here meant the character error rate. In Test-863, the character error rate reduced 3%, which was not so effective as we expected. The insertion and deletion error were also not alleviated too. It might be caused by two factors. First, the weight of sub-optimal path scores in the all-path score was limited. In Equ.(13), the gain of the fusing score by one sub-optimal path was at most ($\beta \cdot \ln 2$), if the sub-optimal path was with an equal score as the optimal path. In practice, the optimal path score was multiple times of the sub-optimal path scores when they were transferred to probabilities. Hence, the gain by combining these sub-optimal paths was limited; the other factor was that the main error in Test-863 was substitution error, which might come from the confusion between models, or be due to the wrong segmentation. Since the insertion and deletion error rates were low in Test-863, the main error

Table 3. The difference between the results achieved by the best path decoding and all-path decoding

Mode	Sub%	Del%	Ins%	Err%
B-A	3.2	0.0	0.1	3.3
S-A*	10.4	0.1	0.1	10.6

owed to the confusion between different models. The current all-path decoding algorithm had done little to reduce such errors.

We had also compared the difference of recognition results obtained by the best path decoding and all-path decoding, as table 3 showed. The recognition hypotheses from one system was taken as the reference for alignment, and the recognition result from the other system was compared with the reference by string comparison algorithm[10]. The higher the character error rate was, the more different between two results was, and vice versa. "B-A" showed the result by taking the "best-path" hypothesis as the reference for alignment with the "all-path" hypothesis. The character error rate was 3.3%, whereas the character error rate difference of two decoding results was only 0.4%, compared with the true transcription file. Hence, the different places between two recognition results should be with low confidence. In order to prove this assertion, we replaced the characters, which were marked as substitution error when the string comparison was done between the two results, with the "Don't Care" symbols[10]. The "Don't Care" symbol can match any one word in the template. Row "S-A*" gave the result obtained by using the true transcription as reference for alignment with the modified "all-path" hypothesis. Ignoring these unstable characters, the substitution error reduced 16.1% and the character error rate reduced 15.1%. Though two decoding results were similar, the differences between two results were informative for low confidential characters. The result achieved by one decoding algorithm was a good complement for the other decoding result.

5 Conclusions

An all-path decoding algorithm, which uses the information of all possible paths or segmentations instead of the best path to recognize the speech utterance, is proposed in this paper. Compared with the conventional best path decoding algorithm, the all-path decoding algorithm has following characteristics,

- The weakness of the dividable assumption is partially amended in decoding by fusing information from all possible paths;
- The decoding result is directly based on Equ.(2), and the best path is not necessary.
- The results obtained by the all-path decoding and the best path decoding are informative and the difference identifies the characters with low confidence.

In above experiments, the all-path decoding algorithm achieved at least a comparable result with that of the best path based algorithm. In tasks dominated by substitution errors, the experiment showed that the approximation by Equ.(3) performed as well as the original scheme, while the all-path decoding algorithm achieved a better performance than the conventional best-path based algorithm in tasks with heavy insertion and deletion errors. Considering the all-path decoding algorithm will not add obvious computation load, it is desirable for SM based decoding.

In this paper, the same acoustic model was taken in both the all-path based decoding and the best path based decoding. The acoustic model was built as the conventional way, that was, it took the dividable assumption during modeling. Such modeling method might introduce mismatch between the model and the decoding algorithm. Our following work will concentrate on building an uniform framework for speech recognition without taking the dividable assumption, that is, the corresponding segments of neighboring acoustic events may be overlapped both in modeling and decoding. We believe it would be useful for a more accurate representation of human speech.

Acknowledgements

This work is supported in part by the China National Nature Science Foundation (No. 60172055, No. 60121302), the Beijing Nature Science Foundation (No.4042025) and the National Fundamental Research Program (973) of China (2004CB318105). The authors would like to thank the anonymous reviewers for their useful comments.

References

1. Gao, S., Lee, T., Wong, Y.W., Xu, B.: Acoustic modeling for chinese speech recognition: A comparative study of mandarin and cantonese. In: Proceedings of International Conference on Acoustics, Speech, and Signal Processing. Volume 3., Istanbul (2000) 1261–1264
2. Ostendorf, M., Roukos, S.: A stochastic segment model for phoneme based continuous speech recognition. IEEE Transactions on Acoustics, Speech and Signal Processing **37**(12) (1989) 1857–1869
3. Ostendorf, M., Digalakis, V., Kimball, O.: From HMM's to segment models: A unified view of stochastic modeling for speech recognition. IEEE Transactions on Speech Audio Processing **4**(5) (1996) 360–378
4. Tang, Y., Liu, W.J., Zhang, H., Xu, B., Ding, G.H.: One-pass coarse-to-fine segmental speech decoding algorithm. In: IEEE Proceedings of International Conference on Acoustics, Speech, and Signal Processing, Toulouse, France (2006) 441–444
5. Tang, Y., Liu, W.J., Zhang, Y.Y., Xu, B.: A framework for fast segment model by avoidance of redundant computation on segment. In: International Symposium on Chinese Spoken Language Processing, Hong Kong (2004) 117–121
6. Tang, Y., Zhang, H., Liu, W.J., Xu, B.: Coloring the speech utterance to accelerate the SM based LVCSR decoding. In: IEEE Proceedings of International Conference on Natural Language Processing and Knowledge Engineering, Wuhan, China (2005) 121–126

7. Rabiner, L., Wilpon, J., Soong, F.: High performance connected digit recognition using hidden markov models. IEEE Transactions on Acoustics, Speech, and Signal Processing **37**(8) (1989) 1214–1225
8. Deng, Y.G., Huang, T.Y., Xu, B.: Towards high performance continuous mandarin digit string recognition. In: Proceedings of the International Conference on Spoken Language Processing, Beijing, China (2000) 642–645
9. Gao, S., Xu, B., Zhang, H., Zhao, B., Li, C.R., Huang, T.Y.: Update of progress of sinohear: Advanced mandarin lvcsr system at NLPR. In: Proceedings of the International Conference on Spoken Language Processing, Beijing, China (2000) 798–801
10. Duda, R., Hart, P., Stork, D.: Pattern Recognition. Second edn. John Wiley & Sons, Inc. (2001)

Improved Mandarin Speech Recognition by Lattice Rescoring with Enhanced Tone Models

Huanliang Wang[1], Yao Qian[2], Frank Soong[2], Jian-Lai Zhou[2], and Jiqing Han[1]

[1] Department of Computer Science and Technology, Harbin Institute of Technology
[2] Microsoft Research Asia, Beijing
{yaoqian, frankkps, jlzhou}@microsoft.com
f-hlwang@msrchina.research.microsoft.com, jqhan@hit.edu.cn

Abstract. Tone plays an important lexical role in spoken tonal languages like Mandarin Chinese. In this paper we propose a two-pass search strategy for improving tonal syllable recognition performance. In the first pass, instantaneous F0 information is employed along with corresponding cepstral information in a 2-stream HMM based decoding. The F0 stream, which incorporates both discrete voiced/unvoiced information and continuous F0 contour, is modeled with a multi-space distribution. With just the first-pass decoding, we recently reported a relative improvement of 24% reduction of tonal syllable recognition errors on a Mandarin Chinese database [5]. In the second pass, F0 information over a horizontal, longer time span is used to build explicit tone models for rescoring the lattice generated in the first pass. Experimental results on the same Mandarin database show that an additional 8% relative error reduction of tonal syllable recognition is obtained by the second-pass search, lattice rescoring with enhanced tone models.

Keywords: tone modeling, lattice rescoring, supra-tone units, tonal syllable recognition.

1 Introduction

Chinese is known as a monosyllabically paced tonal language. Each Chinese character, which is the basic unit in written Chinese, is pronounced as a tonal syllable: a base syllable plus a lexical tone. Correct tonal syllable recognition is critical to differentiate homonyms of same base syllables, e.g., recognizing name of a person or a place and many other application scenarios where strong contextual information or language model is not available in general. Measuring the tonal syllable recognition performance is also a good evaluation of the acoustic model resolution of a recognizer because it is done purely at the phonetic level by removing the language model from the LVCSR decoding process. A recognizer with high tonal syllable recognition rate has many other applications in language learning and objective evaluation of tonal language proficiency of a speaker.

In tonal languages like Chinese, succinct tone modeling is critical for high performance speech recognition due to the lexical nature of tone. The properties of tone, which are listed as follows, have traditionally made tone modeling difficult.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 445 – 453, 2006.

a) Tone is carried by perceivable pitch in the voiced part of a syllable. However, no pitch is perceived in the unvoiced region. As a result, the continuous HMM can not be directly applied to tone modeling since the whole F0 trajectory is discontinuous at the junctures between neighboring voiced and unvoiced segments.

b) Tone is a supra-segmental feature which can span over multiple voiced segments. A time window longer than the window size used for extracting spectral features should be used for extracting tonal features.

There have been studies on tone modeling. A commonly used approach to cope with discontinuous F0 trajectory is to interpolate F0 in the unvoiced segments [1-4]. This approach is obviously incorrect, but a convenient way to bypass the discontinuity problem. The artificially interpolated F0 value has no or even wrong information for identifying a tone. In our recent work [5], we applied a multi-space distribution (MSD) based HMM [6] to tone modeling. It is done without interpolating F0 and resorting to any heuristics. It has been successfully tested in tonal syllable recognition experiments on a Mandarin database. Comparing with tone modeling based on F0 interpolation, the new approach improves absolute tonal syllable error rate by 6%.

Explicit tone modeling [7-8], where tone modeling is separated from spectrum modeling, is commonly employed to take the supra-segmental property of tone into account. The outline feature of a tone contour [9-10] is modeled in such a model. In [10], we proposed supra-tone modeling of Cantonese and used it for Cantonese LVCSR. It characterizes not only the tone contour of a single syllable but also the adjacent bi- or tri-syllables. Experimental results show that the supra-tone modeling outperforms the conventional tone modeling methods significantly.

In decoding, tone (syllable) boundaries are needed for applying explicit tone models. Usually such segmentations can be obtained as a byproduct of the first-pass decoding. However, the optimal tonal syllable sequence might have been pruned prematurely, due to the fact that the tone information is not fully exploited in the search. We propose a two-pass search strategy for improving tonal syllable recognition performance. In [5], we presented our first-pass search with a multi-space distribution (MSD) based tone model and reported corresponding tonal syllable recognition results on a Mandarin Chinese speech database. In this paper, explicit tone modeling is investigated and a lattice rescoring technique with explicit tone models is tested on the same Mandarin database.

The rest of paper is organized as follows. In section 2, we present a two-pass search strategy for improving tonal syllable recognition performance. Explicit tone modeling for Mandarin and lattice rescoring with explicit tone models are studied in section 3. Finally, evaluation experiments are performed on a speaker-independent tonal syllable recognition task in section 4. In section 5 we give a conclusion of this research.

2 Two-Pass Search Strategy with Tone Information

The block diagram of our proposed two-pass search is shown in Figure 1, where the instantaneous F0 information is employed in the first-pass search, while long-term F0 information is used in the second-pass search.

In the first-pass search [5], a tonal syllable lattice is generated with embedded tone information in the acoustic HMMs, in which the tone features and spectral features are separated into two streams and stream-dependent models are constructed to cluster corresponding features into their decision trees. Tone features are modeled by multi-space distribution (MSD) based HMM [6], while spectral features are modeled by conventional HMM.

In the second-pass search, the outline F0 features are firstly extracted with the syllable boundaries given by the first-pass search, and then modeled by explicit tone models. Scores computed from the trained explicit tone models are combined with the scores of tonal syllable obtained in the first-pass search. The combined scores are used to find the best path in the lattice.

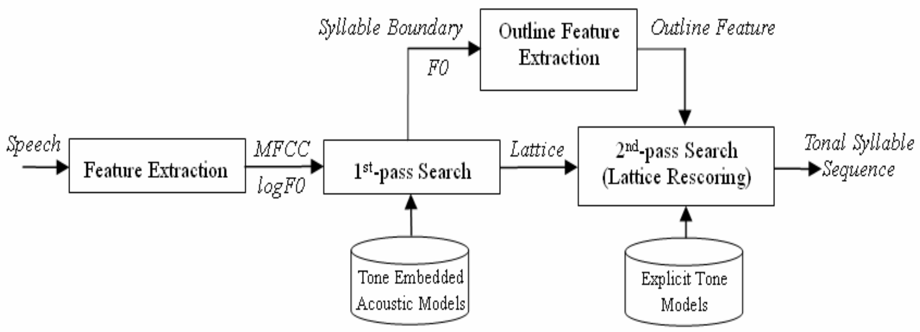


Fig. 1. The block diagram of two-pass search with tone information

3 Lattice Rescoring with Tone Model

3.1 Outline F0 Feature Extraction

F0 is a highly variable acoustic feature. Difference in F0 could be affected by many factors, e.g. age, gender, dialectal difference, health condition, education and idiosyncratic style. Even for the same speaker, the actual range of F0 changes from time to time. Effective F0 normalization is necessary to minimize undesirable fluctuations. In the first-pass search, only logarithm is used to reduce the F0 dynamic range of speakers since search is done time-synchronously and normalization of F0 values on a longer time basis, e.g. sentence, is not feasible. However, in the second-pass search, or lattice rescoring, we can apply utterance-based normalization to F0, which is defined as

$$\tilde{F}_i = \frac{F_i}{mean(F_i)} \tag{1}$$

where $\{F_i\}_{i=1}^{N}$ is a sequence of original F0 values and N is the total number of voiced frames in that utterance. F0 normalization in the above equation (1) can reduce F0 variation at both inter- and intra-speaker levels.

Lattice rescoring can integrate high-level knowledge sources that may not be easily incorporated in the first decoding pass, e.g. long-term F0 information. We extract outline F0 features for tone modeling. The tone contour of a syllable is evenly divided into three sections, and each section is represented by the corresponding F0 mean of all F0 values in that section. Figure 2 shows an example of outline F0 feature extraction for the tonal syllable "pu3 tong1 hua4". The outline F0 feature is a rough sketch of the F0 contour. The window for feature extraction is whole syllable, the carrier of the tone. The outline F0 features indicate an averaged pitch trajectory.

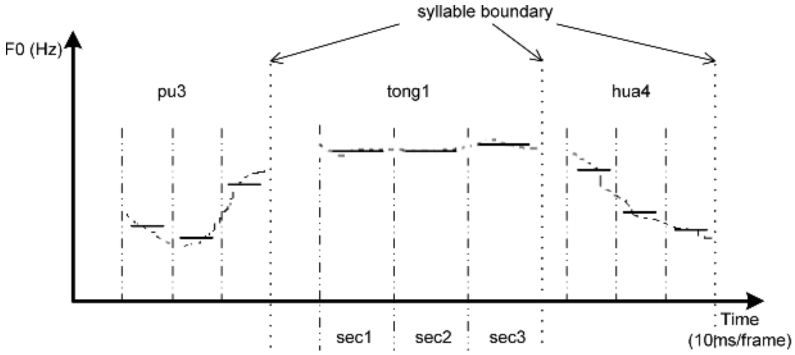


Fig. 2. An example of outline F0 feature extraction for a tonal syllable sequence "pu3 tong1 hua4"

3.2 Supra-Tone Modeling

In continuous speech, the F0 contour of the current tone can be affected significantly by the neighboring tones due to co-articulation. Supra-tone modeling [10] characterizes not only the tone contour of the current syllable but also the adjacent ones. Supra-tone model has been shown to outperform the conventional tone models in Cantonese tone recognition. Here, we apply it to Mandarin.

A supra-tone unit covers the contour of successive tones. Figure 3 shows di-tone units of tonal syllable sequence "pu3 tong1 hua4". The di-tone unit to be modeled covers two consecutive tones and there is an overlap of one syllable between two adjacent di-tone units. In this way, the supra-tone unit characterizes long-term, inter-syllabic speech dynamics in both time and frequency. Supra-tone modeling is different from context-dependent tone modeling, which is based on phonetic context rather than acoustic context. The supra-tone models capture the tonal context effect by using not only the tone identities but also the acoustic features.

Gaussian mixture model (GMM) is employed to model the supra-tone units. Theoretically, GMM with sufficient mixture components is capable of approximating any arbitrary statistical distribution. Moreover, GMM provides a probabilistic output that can be readily integrated into HMM based ASR system.

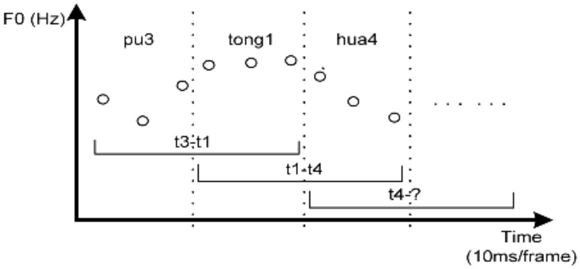


Fig. 3. Di-tone units of tonal syllable “pu3 tong1 hua4”

3.3 Lattice Rescoring

Supra-tone model is used to rescore the tonal syllable lattice. The search process in the lattice is rewritten as

$$TS^* = \underset{TS}{\arg\max}(\alpha \log P(TS \mid O_{TS}) + \beta \log P(T \mid O_T) + \gamma P(TS) + WP) \qquad (2)$$

where TS is a sequence of tonal syllables, $P(TS \mid O_{TS})$ is the score of a tonal syllable sequence obtained in the first-pass search, $P(T \mid O_T)$ is the score obtained by evaluating the outline F0 features against a supra-tone model, $P(TS)$ is the prior probability of a tonal syllable sequence, and WP is the word penalty. α, β and γ are the weights of corresponding models and they are optimized in a development set. In our tonal syllable recognition task, a free tonal syllable loop is used as language model and α is set to one. Therefore, only β and WP need to be estimated. A Viterbi search is employed to find the best tonal syllable sequence in the lattice.

4 Experimental Results and Analysis

4.1 Experimental Setup

The evaluation experiments are performed on a speaker-independent database of read speech. Training set consists of 50k utterances (80 hours) data recorded by 250 male speakers. Development set consists of 5 speakers and 100 utterances (about 2k tonal syllables). Testing set consists of 400 utterances (about 8k tonal syllables) from 20 speakers. Speakers in the training, development and testing sets are all different. For tone recognition experiments, we use all 500 utterances from the development and testing sets as the testing set. In all following experiments, a Mandarin tone is classified as one of five categories, i.e. tone 1 to tone 5.

The performance of the first-pass search

In the first-pass search, MSD-HMM based tri-phone models are used as tone embedded acoustic models, which have totally 5,000 tied states and 16 Gaussian components/state. Acoustic features are made of 39-dimensional MFCC and

5-dimensional, extended F0 feature. Those two sets of features are divided into two streams. Free tonal syllable loop (i.e. no language model is used) is employed in the decoding and a tonal syllable lattice is generated by the recognizer for each utterance.

The performances of tonal syllable and tone recognition in the first-pass search are shown in Table 1, where the second line is the one-best result and third line is the graph error rate (GER). GER is computed by aligning the correct tonal syllable sequence with the generated lattice to find the path with the least number of tonal syllable errors. The error rates of tonal syllable are 35.0% and 10.4% in one-best and lattice, respectively. The large difference between the two tonal syllable error rates indicates that by rescoring tonal syllable lattice with high-level knowledge sources like long term F0 information we might be able to improve the recognition performance significantly.

Table 1. The performances of one-best results and generated lattice on the testing set

	Tonal syllable error rate (TSER)	Tone error rate (TER)
One-best (%)	35.0	24.9
Lattice (%)	10.4	7.81

Experimental setup for the second-pass search
GMM is used to model the supra-tone units and each model has 32 Gaussian components with full covariance. The training data is force-aligned by the first-pass recognizer to get the syllable boundaries for tone modeling. Outline features of F0 contour and dynamic F0 contour are evaluated for supra-tone modeling. In order to make comparison with the conventional tone modeling method, MSD-HMM is used to explicitly model tone in the following tone recognition experiments. Each MSD-HMM consists of 4 states with a left-to-right topology. Each state is made up of 32 Gaussian components.

4.2 Role of F0 Normalization

We compare the performance of two different F0 normalizations with the original F0 on a tone recognition experiment, where mono-tone based supra-tone model is used. Table 2 shows the experimental results.

Table 2. Tone recognition results using different F0 normalization methods

	Original F0	Logarithm normalization	Utterance-based normalization
Tone Error Rate (%)	39.1	38.7	38.3

From Table 2 we can see that utterance-based normalization method yields the best tone recognition performance. In the following experiments, utterance-based normalization method is used.

4.3 The Experiments of Tone Recognition

Quality tone models are important for improving lattice rescoring performance. In order to evaluate tone model resolution, tone recognition experiments are performed on the testing set. The tone recognition results of three supra-tone models with/without dynamic outline F0 feature are shown in Figure 4, where

1) GMM-ST-MT: mono-tone based supra-tone model
2) GMM-ST-DT: di-tone based supra-tone model
3) GMM-ST-TT: tri-tone based supra-tone model

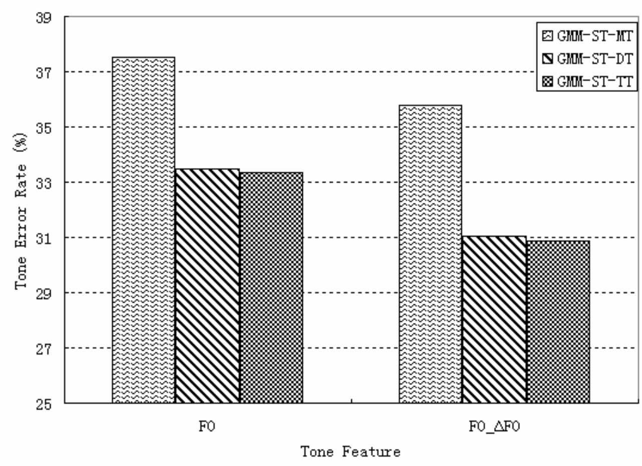


Fig. 4. Tone recognition results of three supra-tone models with/without dynamic outline F0 feature

and F0_ΔF0 is the case with dynamic outline F0 feature which is the mean of delta F0 in each section of the F0 contour. Figure 4 shows that the supra-tone units (di-tone and tri-tone) and dynamic outline F0 feature can significantly improve the performance of tone recognition.

In order to compare the performance of supra-tone model with other tone modeling methods, the following two experiments are performed.

1) Conventional context-dependent tone modeling with outline F0 feature
2) MSD-based HMM for tone modeling

The experimental results show that supra-tone modes with outline F0 feature can outperform above 1) and 2) models by about 3% and 4% absolute tone error rate reduction, respectively. It indicates that the outline F0 feature and a wider time window are benefit to Mandarin tone recognition.

4.4 The Experiments of Lattice Rescoring with Supra-Tone Model

Di-tone models with/without dynamic outline F0 feature are used in lattice rescoring. Tri-tone models are not tried in these experiments since their performances of tone recognition are only slightly better than those of di-tone models while their computation complexities are much higher. The optimal weight β and *WP* for computing the new search score in Equation 2) are found on the development data set with a full grid search. For di-tone model without dynamic outline F0 features, total tonal syllable error rates of lattice rescoring with different tone model weights and word penalties on the development set are shown via a contour plot in Figure 5. It indicates that the optimal weight pair is (4.7, -49). Rescoring results on the testing data set with optimal weights are shown in Table 3. It shows that the di-tone models with dynamic outline F0 feature can achieve 8% relative TSER reduction and 12.4% relative TER reduction.

Table 3. Results of lattice rescoring using di-tone models with/without dynamic outline F0 feature

Tone Model Type	Optimal point (β,*WP*)	TSER (%)	TER (%)
Baseline	(0,-35)	35.0	24.9
GMM-ST-DT_F0	(4.7, -49)	32.4	22.5
GMM-ST-DT_F0-ΔF0	(3.5, -67)	32.2	21.8

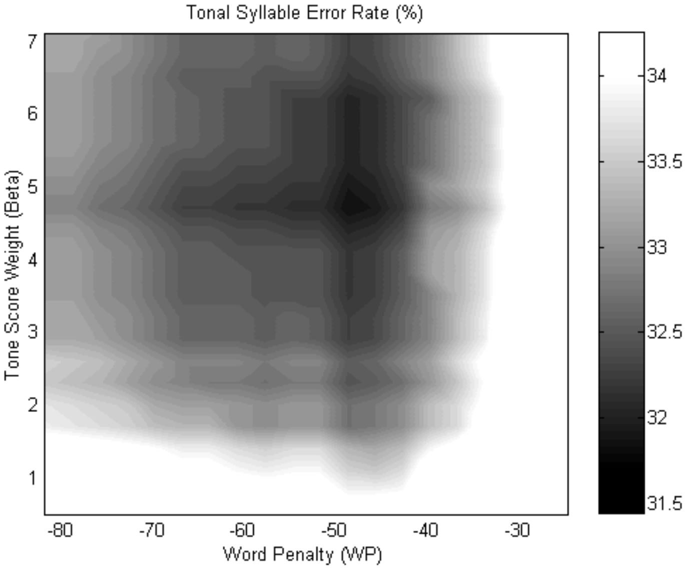


Fig. 5. Contour plot of tonal syllable error rate for different tone model weights and word penalties

5 Conclusions

In the paper, a two-pass search strategy with tone information is introduced for improving tonal syllable recognition performance. Lattice rescoring with enhanced tone models is particularly investigated. Explicit tone modeling with outline F0 features via supra-tone model is applied to Mandarin Chinese. In the experiments of tone recognition, supra-tone modeling shows improved performance over the conventional tone modeling methods. In lattice rescoring, by using di-tone models with dynamic outline F0 feature we obtains an 8% tonal syllable error reduction and 12.4% relative tone error reduction all relative, compared to the single-best results in the first-pass search.

References

1. Hirst, D., Espesser, R.: Automatic Modeling of Fundamental Frequency Using a Quadratic Spline Function. Travaux de l'Institut de Phonétique d'Aix 15, (1993) 71-85
2. Chen, C.J., Gopinath, R.A., Monkowski, M.D., Picheny, M.A., Shen, K.: New Methods in Continuous Mandarin Speech Recognition. In Proc. Eurospeech 1997, (1997) 1543-1546
3. Chang, E., Zhou, J.-L., Di, S., Huang, C., Lee, K-F.: Large Vocabulary Mandarin Speech Recognition with Different Approach in Modeling Tones. In Proc. ICSLP 2000, (2000) 983-986
4. Freij, G.J., Fallside, F.: Lexical Stress Recognition Using Hidden Markov Models. In Proc. ICASSP 1988, (1988) 135-138
5. Wang, H.L., Qian, Y., Soong, F.K., Zhou, J.-L., Han, J.Q.: A Multi-Space Distribution (MSD) Approach to Speech Recognition of Tonal Languages. In Proc. ICSLP 2006
6. Tokuda, K., Masuko, T., Miyazaki, N., Kobayashi, T.: Multi-space Probability Distribution HMM. IEICE Trans. Inf. & Syst., E85-D(3), (2002) 455-464
7. Lin, C.H., Wu, C.H., Ting, P.Y., Wang, H.M.: Framework for Recognition of Mandarin Syllables with Tones Using Sub-syllabic Units. Journal of Speech Communication, Vol.18, No.2, (1996) 175-190
8. Qian, Y., Soong, F.K., Lee, T.: Tone-enhanced Generalized Character Posterior Probability (GCPP) for Cantonese LVCSR. In Proc. ICASSP 2006, (2006) 133-136
9. Tian, Y., Zhou, J.-L., Chu, M., Chang, E.: Tone Recognition with Fractionized Models and Outlined Features. In Proc. ICASSP 2004, (2004) 105-108
10. Qian, Y.: Use of Tone Information in Cantonese LVCSR Based on Generalized Character Posterior Probability Decoding. PhD. Thesis, CUHK, 2005

On Using Entropy Information to Improve Posterior Probability-Based Confidence Measures

Tzan-Hwei Chen[1], Berlin Chen[1], and Hsin-Min Wang[2]

[1] Graduate Institute of Computer Science & Information Engineering,
National Taiwan Normal University, Taipei
{g93470018, berlin}@csie.ntnu.edu.tw
[2] Institute of Information Science Academia Sinica, Taipei
whm@iis.sinica.edu.tw

Abstract. In this paper, we propose a novel approach that reduces the confidence error rate of traditional posterior probability-based confidence measures in large vocabulary continuous speech recognition systems. The method enhances the discriminability of confidence measures by applying entropy information to the posterior probability-based confidence measures of word hypotheses. The experiments conducted on the Chinese Mandarin broadcast news database MATBN show that entropy-based confidence measures outperform traditional posterior probability-based confidence measures. The relative reductions in the confidence error rate are 14.11% and 9.17% for experiments conducted on field reporter speech and interviewee speech, respectively.

Keywords: confidence measure, entropy, posterior probability, continuous speech recognition.

1 Introduction

With the growing number of applications for automatic speech recognition (ASR) systems, the robustness and stability of a speech recognizer has become increasing important. The performance of ASR systems in real-world applications usually degrades dramatically compared to that of laboratory ASR systems. Therefore, verifying the recognition output of ASR systems is a critical issue. Confidence measures can be used to automatically label individual hypothesized words in the output of ASR systems as either *correct* or *incorrect*. This additional appraisal of word sequences has been adopted in unsupervised model training [1], and to improve the recognition accuracy [2].

Confidence measure algorithms can be roughly classified into three major categories [3] as follows:

1) Feature-based: These approaches assess the confidence based on some selected features, such as word duration, acoustic score, language model back-off, and part-of-speech.
2) Explicit model-based: These approaches treat confidence measures as hypothesis testing problems [4], and need to model extra alternative hypotheses.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 454 – 463, 2006.

3) Posterior probability-based: The posterior probability estimated according to the standard Maximum *a Posteriori* (MAP) framework is a good candidate for confidence measures, because it has a good bounded range between 0 and 1. The superior performance of the posterior probability has been demonstrated by using it as a confidence measure [5] [6].

In this paper, our objective is to improve the posterior probability-based approach by integrating entropy information into the posterior probability-based confidence measures of word hypotheses using a word graph. The experiments conducted on the Chinese Mandarin broadcast news database MATBN show that our approach can effectively reduce the confidence error rate.

The remainder of the paper is organized as follows. Section 2 describes traditional posterior probability-based confidence measures [5]. In Section 3, we explain how to combine the entropy information with the posterior probability-based confidence measures. Section 4 introduces the ASR system and the databases used in this paper. The experiment results are detailed in Section 5. Finally, in Section 6, we present our conclusions and indicate some future research directions.

2 Traditional Posterior Probability-Based Confidence Measures

The fundamental rule in statistical speech recognition systems tries to find a word sequence $\{[w]_1^M\}_{opt}$ that maximizes the posterior probability, given a sequence of acoustic observations *X* of length *T*:

$$\begin{aligned}\{[w]_1^M\}_{opt} &= \arg\max_{[w]_1^M} p([w]_1^M \mid X) \\ &= \arg\max_{[w]_1^M} \frac{p(X \mid [w]_1^M)p([w]_1^M)}{p(X)}, \\ &= \arg\max_{[w]_1^M} p(X \mid [w]_1^M)p([w]_1^M)\end{aligned} \tag{1}$$

where $p([w]_1^M)$ and $p(X \mid [w]_1^M)$ denote the language model probability and the acoustic model probability, respectively; and $p(X)$ represents the prior probability of the acoustic observation sequence X . In practical implementations, the denominator term $p(X)$ is omitted because it is invariant for all possible word sequences. Apart from being used to select the most likely sentence, the sentence posterior probability also serves as a good confidence measure. However, the following question arises: How can we compute the sentence posterior probability?

2.1 Calculating the Posterior Probability of a Hypothesized Word

Given a hypothesized sentence (or word sequence), the posterior probability of a hypothesized word $[w,s,e]$[1] in the sentence, $p([w,s,e] \mid X)$, is equal to the sum of the posterior probabilities of all the sentences that contain the word:

[1] *s* is the start time and *e* is the end time of the hypothesized word *w*.

$$p([w,s,e]\,|\,X)=\frac{\displaystyle\sum_{\substack{[w_m,s_m,e_m]_{m=1}^{M}\\ w_m=w,s_m=s,e_m=e}}\left\{\prod_{m=1}^{M} p(X_{s_m}^{e_m}\,|\,w_m)\cdot p(w_m\,|\,h_m)^{\kappa}\right\}}{\displaystyle\sum_{[w_n,s_n,e_n]_{n=1}^{N}\in \mathbf{W}_{\Sigma}}\left\{\prod_{n=1}^{N} p(X_{s_n}^{e_n}\,|\,w_n)\cdot p(w_n\,|\,h_n)^{\kappa}\right\}}. \tag{2}$$

In Eq. (2), $p(X_{s_m}^{e_m}\,|\,w_m)$ is the acoustic likelihood; $p(w_m\,|\,h_m)$ is the language model probability given the preceding history h_m; $\mathbf{W}_{\Sigma}$ denotes the set of all possible word sequences belonging to the language; N denotes the number of words in an arbitrary word sequence; and κ is the language model scaling factor. Note that the denominator term in Eq. (2) is equal to $p(X)$ in Eq. (1). Practically speaking, it is infeasible to enumerate all the word sequences in the implementation of a speech recognition system. Thus, to calculate the denominator term in Eq. (2), a word graph is usually adopted to approximate the word sequence set $\mathbf{W}_{\Sigma}$. Let Ψ^{X} denote the word graph generated for an acoustic observation sequence X. The posterior probability of the hypothesized word $[w,s,e]$, $p([w,s,e]\,|\,X)$, can be approximated as $p(a:[w,s,e]\,|\,\Psi^{X})$ computed by:

$$p(a:[w,s,e]\,|\,\Psi^{X})=\frac{\displaystyle\sum_{\substack{[w_m,s_m,e_m]_{m=1}^{M}\in\Psi^{X}\\ w_m=w,s_m=s,e_m=e}}\left\{\prod_{m=1}^{M} p(X_{s_m}^{e_m}\,|\,w_m)\cdot p(w_m\,|\,h_m)^{\kappa}\right\}}{\displaystyle\sum_{[w_n,s_n,e_n]_{n=1}^{N}\in \Psi^{X}}\left\{\prod_{n=1}^{N} p(X_{s_n}^{e_n}\,|\,w_n)\cdot p(w_n\,|\,h_n)^{\kappa}\right\}}, \tag{3}$$

where $a:[w,s,e]$ denotes the word arc associated with the hypothesized word $[w,s,e]$ in the word graph. The word posterior probability can be computed efficiently by applying a forward-backward algorithm to the word graph [5]. The forward score $\alpha(a:[w,s,e])$ is recursively computed from the start time of the word graph to the start time of the word arc $a:[w,s,e]$, i.e., s:

$$\alpha(a:[w,s,e])=p(X_s^e\,|\,w)\times\sum_{a':[w',s',s-1]}\alpha(a':[w',s',s-1])p(w\,|\,h_w)^{\kappa}, \tag{4}$$

where the summation is conducted for all word arcs ending at $s-1$; $p(X_s^e\,|\,w)$ is the acoustic likelihood; and h_w is the preceding history of w. Similarly, a backward score $\beta(a:[w,s,e])$ is calculated from the end time of the word graph to the end time of $a:[w,s,e]$, i.e., e:

$$\beta(a:[w,s,e])=\sum_{a':[w',e+1,e']}\beta(a':[w',e+1,e'])p(w'\,|\,w)^{\kappa}p(X_{e+1}^{e'}\,|\,w'), \tag{5}$$

where the summation is conducted for all word arcs starting at $e+1$. Obviously, the numerator in Eq. (3) is the product of $\alpha(a:[w,s,e])$ and $\beta(a:[w,s,e])$, while the denominator in Eq. (3) is the summation of the forward scores of all end-word arcs in the word graph. Therefore, Eq. (3) can be rewritten as:

$$p(a:[w,s,e]\,|\,\Psi^X) = \frac{\alpha(a:[w,s,e])\times\beta(a:[w,s,e])}{\sum_{a':[w',s',T]\in\Psi^X}\alpha(a':[w',s',T])}. \tag{6}$$

2.2 Posterior Probability-Based Confidence Measure of a Hypothesized Word

The word arc posterior probability calculated by Eq. (6) can be used directly as a measure of confidence for the word hypothesis $[w,s,e]$:

$$C_{normal}([w,s,e]) = p(a:[w,s,e]\,|\,\Psi^X). \tag{7}$$

Consider a word arc $a:[w,s,e]$ in a word graph. There usually exist some alternative word arcs whose word identities are identical to *w*, but the time marks are slightly different to [*s*,*e*]. The more such alternative word arcs exist, the more likely it is that [*w*,*s*,*e*] is correct and should be accepted. Based on this concept, Wessel *et al.* proposed three methods for calculating the confidence of a hypothesized word $[w,s,e]$ according to the word arc posterior probabilities [5]:

$$C_{\sec}([w,s,e]) = \sum_{\substack{a':[w,s',e'] \\ \{s',\ldots,e'\}\cap\{s,\ldots,e\}\neq\phi}} p(a':[w,s',e']\,|\,\Psi^X), \tag{8}$$

$$C_{med}([w,s,e]) = \sum_{\substack{a':[w,s',e'] \\ s'\le(s+e)/2\le e'}} p(a':[w,s',e']\,|\,\Psi^X), \tag{9}$$

and

$$C_{\max}([w,s,e]) = \max_{t\in\{s,\ldots,e\}} \sum_{\substack{a':[w,s',e'] \\ s'\le t\le e'}} p(a':[w,s',e']\,|\,\Psi^X). \tag{10}$$

In Eq. (8), the summation is over all word arcs containing the same word identity that intersect with the word arc $a:[w,s,e]$. In Eq. (9), only word arcs that associate with the same word identity *w* and intersect the median time of the word arc $a:[w,s,e]$ are considered. In Eq. (10), for each time instance between *s* and *e*, the posterior probabilities of the word arcs whose word identities are *w* are accumulated. This process yields *e*-*s*+1 accumulated posterior probabilities. Then, the confidence of the word hypothesis [*w*,*s*,*e*] is the maximum of these accumulated posterior probabilities.

3 The Proposed Approach

To determine whether a recognized word is correct or not, it might be helpful to take all the other word arcs with time boundaries similar to the target word arc hypothesis

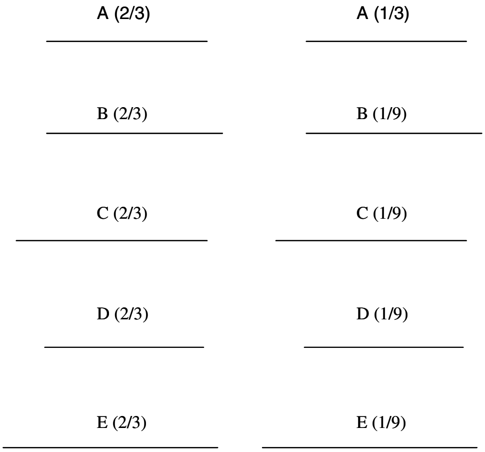


Fig. 1. Two examples of word arcs extracted from word graphs. The values in parentheses represent the confidence (e.g., C_{max}) of the word arc.

into account, instead of just considering the word arcs whose word identities are identical to the recognized word to be evaluated. As illustrated in Fig. 1, the confidence of word 'A' on the left-hand side is not reliable in terms of alternative words because all the confidence measures are high. In contrast, the confidence of word 'A' on the right-hand side is trustworthy because it is relatively higher than those of the other words. Entropy, described as "a measure of the disorder", is a way to measure the amount of information in a random variable. To emphasize the reliability of confidence measure, we propose an entropy-based approach that evaluates the degree of confusion in confidence measures. By incorporating entropy information into traditional posterior probability-based confidence measures, the new entropy-based confidence measure of a hypothesized word is defined as:

$$C_{entropy}([w,s,e]) = CM([w,s,e]) \cdot (1 - E_{avg}([w,s,e])), \tag{11}$$

where $CM([w,s,e])$ denotes a traditional confidence measure of $[w,s,e]$, which can be estimated by Eqs. (7), (8), (9), or (10); and $E_{avg}([w,s,e])$ is the average normalized entropy defined as:

$$E_{avg}([w,s,e]) = \frac{1}{e-s+1} \sum_{t=s}^{e} E_f(t). \tag{12}$$

The larger the $E_{avg}([w,s,e])$, the greater the degree of uncertainty there will be about the confidence measure. Consequently, the originally estimated confidence of a recognized word is considered more unreliable. Based on this concept, we weight the conventional confidence measure by $1 - E_{avg}([w,s,e])$. In Eq. (12), $E_f(t)$ is computed by,

$$E_f(t) = -\frac{1}{\log_2 N} \sum_{[w,s,e],s \le t \le e} P_{CM}([w,s,e],t) \log_2 P_{CM}([w,s,e],t), \tag{13}$$

where N is the number of distinct word identities in frame *t*; and $P_{CM}([w,s,e],t)$ represents the normalized confidence in each frame *t*, calculated by

$$P_{CM}([w,s,e],t) = \frac{CM_{sum}([w,s,e],t)}{\sum_{[w',s',e'],s' \le t \le e'} CM_{sum}([w',s',e'],t)}, \tag{14}$$

and

$$CM_{sum}([w,s,e],t) = \sum_{\substack{a':[w,s',e'] \\ s' \le t \le e'}} CM([w,s',e'],t). \tag{15}$$

When computing the entropy, we only consider the distribution of different words. We take the summation over the confidence of words with the same identity before calculating $P_{CM}([w,s,e],t)$. In [7], the entropy information was considered as one of predictor features in a confusion network. In this paper, we integrate entropy information into the posterior probability-based confidence measures directly.

4 The Speech Recognition System

The large vocabulary continuous speech recognition (LVCSR) system [1] and the databases used in this paper are described in this section.

4.1 Front-End Signal Processing

Front-end signal processing is performed with the HLDA-based [8] data-driven Mel-frequency feature extraction approach, and further processed by MLLT transformation for feature de-correlation. Finally, utterance-based feature mean subtraction and variance normalization are applied to all the training and test utterances.

4.2 Speech Corpus and Acoustic Model Training

In this work, we use the MATBN (Mandarin Across Taiwan Broadcast News) speech database [9], which was collected by Academia Sinica and Public Television Service Foundation of Taiwan between November 2001 and April 2003. Approximately 200 hours of speech data was supplemented with corresponding orthographic transcripts. Our experiments are conducted on the “field reporter” and “interviewee” subsets. The statistics of these two subsets are summarized in Table 1.

The acoustic models used in our LVCSR system are comprised of 112 right-context-dependent INITIAL models, 38 context-independent FINAL models, and a silence model. The models are first trained by using the Baum-Welch training algorithm according to the ML criterion, and then optimized by the MPE-based [10] discriminative training approach.

Table 1. The statistics of two speech data sets used in this paper

	Field reporter speech	Interviewee speech
Training data	25.5 h	8.80 h
Validation data	0.74 h	0.45 h
Test data	1.5 h	0.60 h

4.3 The Lexicon and Language Model

The recognition lexicon consists of 72K words. The language models used in this paper consist of trigram and bigram models. They were estimated from a text corpus of 170 million Chinese characters collected from Central News Agency in 2001 and 2002 (the Chinese Gigaword Corpus released by LDC) based on the ML criterion. For the interviewee task, we used the Mandarin Conversational Dialogue Corpus (MCDC) [11] to train an in-domain language model. The *n*-gram language models were trained with the SRI Language Modeling Toolkit (SRILM) [12]. We also employed the Katz back-off smoothing technique.

4.4 Speech Recognition

The speech recognizer is implemented with a left-to-right frame-synchronous Viterbi Tree search and a lexical prefix tree organization of the lexicon. The recognition hypotheses are organized into a word graph for further language model rescoring. In this study, the word bigram language model is used in the tree search procedure, while the trigram language model is used in the word graph rescoring procedure [1].

5 Experiments

5.1 Experiment Setup

The acoustic models for the field reporter and interviewee tasks were trained with approximately 25 hours and 9 hours of speech, respectively. The acoustic models for field reporters were trained by 150 iterations of ML training and 10 iterations of MPE training; while the acoustic models for interviewees were trained by 30 iterations of ML training and 8 iterations of MPE training. The character error rates of the two tasks are shown in Table 2.

Table 2. Recognition results for the two test subsets

	Field reporter speech	Interviewee speech
Character error rate	20.79%	49.56%

5.2 Evaluation Metric

The performance of the confidence measure is evaluated on the basis of the confidence error rate (CER) defined as:

$$CER = \frac{\text{\# falsely accepted words} + \text{\# falsely rejected words}}{\text{\# recognized words}} . \qquad (16)$$

CER can be clarified as follows: Given the confidence of a hypothesized word and a rejection threshold, the word is labeled *correct* (i.e., accepted) or *incorrect* (i.e., rejected). If an incorrectly recognized word is labeled *correct*, it is a false acceptance; similarly, if a correctly recognized word is tagged *incorrect*, it is a false rejection. The baseline CER is calculated as the number of insertions and substitutions divided by the number of recognized words. Obviously, the CER is heavily dependent on the choice of the rejection threshold. In our experiments, the threshold was adjusted to minimize the CER of the validation set. Then, the threshold that yielded the minimal CER for the validation set was applied to the test set.

Another evaluation metric is the detection-error-tradeoff (DET) curve, which contains a plot of the false acceptance rate over the false rejection rate for different thresholds.

5.3 Experiment Results

The comparison of the proposed entropy-based confidence measures and the traditional posterior probability-based confidence measures is shown in Table 3. The language model scaling factor κ in Eq. (3) is set to 11. The third to sixth rows in the table show the results obtained when Eqs. (7), (8), (9) and (10) are used, respectively, and the seventh to tenth rows are the results obtained when entropy information is integrated into the listed methods. From Table 3, it is clear that the proposed entropy-based confidence measures outperform traditional posterior probability-based confidence measures. The proposed approach achieves a relative CER reduction of 14.11% over the traditional approach ($C_{entropy}(C_{sec})$ or $C_{entropy}(C_{med})$ versus C_{sec}, C_{med}, or C_{max}) in the field reporter task; and a relative reduction of 9.17% ($C_{entropy}(C_{med})$ versus C_{normal}) in the interviewee task.

The DET curves of C_{max} and $C_{entropy}(C_{sec})$ for the field reporter task and the DET curves of C_{normal} and $C_{entropy}(C_{med})$ for the interviewee task are shown in Figs. 2 and 3, respectively. Again, we find that the proposed methods outperform traditional methods.

Table 3. Experiment results using the confidence error rate to evaluate traditional posterior probability-based confidence measures and entropy-based confidence measures

Methods	Field reporter speech	Interviewee speech
baseline	24.52%	51.97%
C_{normal}	22.18%	31.31%
C_{sec}	21.47%	32.32%
C_{med}	21.47%	32.32%
C_{max}	21.47%	32.32%
$C_{entropy}(C_{normal})$	18.55%	31.08%
$C_{entropy}(C_{sec})$	18.44%	28.50%
$C_{entropy}(C_{med})$	18.44%	28.44%
$C_{entropy}(C_{max})$	18.45%	28.51%

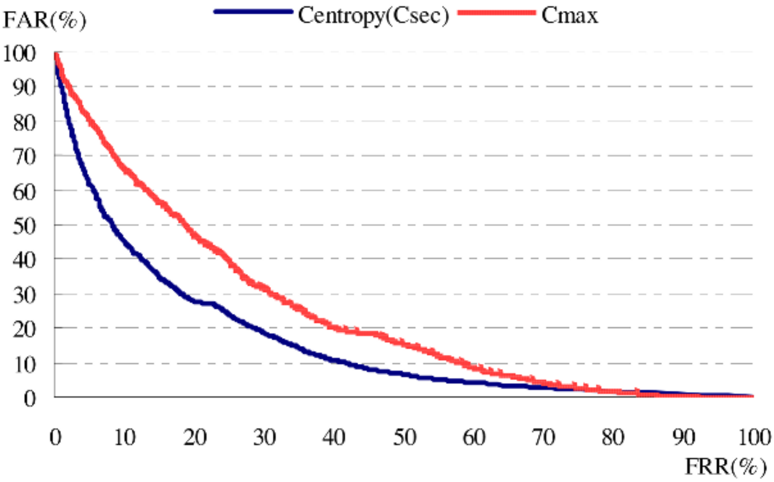


Fig. 2. DET curves for the test set of the field reporter task

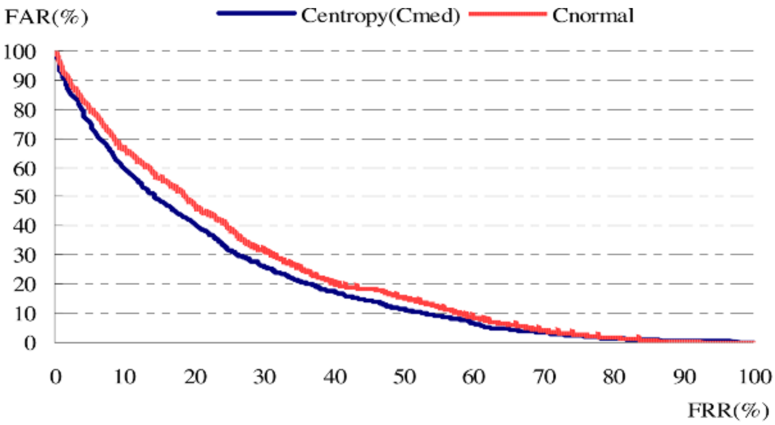


Fig. 3. DET curves for the test set of the interviewee task

6 Conclusions

We have presented a new approach that combines traditional posterior probability-based confidence measures with entropy information to verify the output of large vocabulary continuous speech recognition systems. The proposed methods were evaluated on two speech recognition tasks: a field reporter speech set and an interviewee speech set. In the field reporter speech set, the proposed methods achieved a 14.11% relative reduction in the confidence error rate compared to traditional methods, while in the interviewee speech set, the proposed methods achieved a 9.17% relative reduction. In our future work, we will incorporate the entropy information into feature-based confidence measures or other confidence measures.

References

1. B. Chen, J.-W. Kuo and W.-H. Tsai, “Lightly Supervised and Data-driven Approaches to Mandarin Broadcast News Transcription,” International Journal of Computational Linguistics and Chinese Language Processing, Vol. 10, No. 1, pp.1-18, 2005.
2. F. Wessel, R. Schlüter and H. Ney, “Using Posterior Probabilities for Improved Speech Recognition,” in Proc. ICASSP 2000.
3. W. K. Lo and F. K. Soong, “Generalized Posterior Probability for Minimum Error Verification of Recognized Sentences,” in Proc. ICASSP 2005.
4. R. C. Rose, B.-H. Juang and C.-H. Lee, “A Training Procedure for Verifying String Hypothesis in Continuous Speech Recognition,” in Proc. ICASSP 1995.
5. F. Wessel, R. Schlüter and H. Ney, ”Confidence Measures for Large Vocabulary Continuous Speech Recognition,” IEEE Trans. Speech and Audio Processing, Vol. 9, No. 3, pp.288-298, 2001.
6. W. K. Lo, F. K. Soong and S. Nakamura, “Generalized Posterior Probability for Minimizing Verification Errors at Subword, Word and Sentence Levels,” in Proc. ISCSLP, 2004.
7. J. Xue and Y. Zhao, “Random Forests-based Confidence Annotation Using Novel Features from Confusion Network,” in Proc. ICASSP 2006.
8. H. J. F. Gales, “Semi-tied Covariance Matrices for Hidden Markov Models,” IEEE Trans. on Speech and Audio Processing, Vol. 7, No. 3, pp. 272-281, 1999.
9. H.-M. Wang, B. Chen, J.-W. Kuo, and S.-S Cheng, “MATBN: A Mandarin Chinese Broadcast News Corpus,” International Journal of Computational Linguistics and Chinese Language Processing, Vol. 10, No. 2, pp. 219-236, 2005.
10. D. Povey, “Discriminative Training for Large Vocabulary Speech Recognition,” Ph. D Dissertation, Peterhouse, University of Cambridge, 2004.
11. S. C. Tseng and Y. F. Liu, “Mandarin Conversational Dialogue Corpus,” MCDC. Technical Note 2001-01, Institute of Linguistics, Academia Sinica, Taipei.
12. A. Stolcke, SRI language Modeling Toolkit version 1.3.3, http://www.speech.sri.com/projects/srilm/.

Vietnamese Automatic Speech Recognition: The FLaVoR Approach

Quan Vu, Kris Demuynck, and Dirk Van Compernolle

K.U.Leuven/ESAT/PSI
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
qvuhai@esat.kuleuven.be
www.esat.kuleuven.be/psi/spraak/

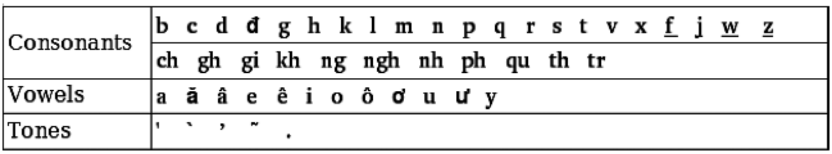
Abstract. Automatic speech recognition for languages in Southeast Asia, including Chinese, Thai and Vietnamese, typically models both acoustics and languages at the syllable level. This paper presents a new approach for recognizing those languages by exploiting information at the word level. The new approach, adapted from our FLaVoR architecture[1], consists of two layers. In the first layer, a pure acoustic-phonemic search generates a dense phoneme network enriched with meta data. In the second layer, a word decoding is performed in the composition of a series of finite state transducers (FST), combining various knowledge sources across sub-lexical, word lexical and word-based language models. Experimental results on the Vietnamese Broadcast News corpus showed that our approach is both effective and flexible.

Keywords: Automatic speech recognition, finite state transducer, phoneme network, Vietnamese.

1 Introduction

Like Chinese [5], Thai [4] and other languages in Southeast Asia, Vietnamese is a tonal, morpho-syllabic language in which each syllable is represented by a unique word unit (WU) and most WUs are also morphemes, except for some foreign words, mainly borrowed from English and French. Notice that the term WU we use here has a similar meaning to the term "character" in Chinese. There are six distinct tones and around seven thousand WUs found in the language. Of these 7000, about five thousand WUs are frequently used [2]. The Vietnamese writing system, on the other hand, is completely different from the ones in Southeast Asia, including Chinese. In fact, it is based on an extended Latin symbol set as given in Fig 1. The underlined symbols in Fig 1 are not in the original system. In addition, the last five symbols are tone marks (the sixth tone has no mark). Fig 2 shows some examples of Vietnamese WUs and words. This example sequence contains six WUs and three words. With this writing system, even if WUs are separated by spaces, the problem of word segmentation is not trivial. As with Chinese, words are not well defined. Each word is composed of one to several WUs with different meaning.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 464–474, 2006.



Consonants	b c d đ g h k l m n p q r s t v x f j w z
	ch gh gi kh ng ngh nh ph qu th tr
Vowels	a ă â e ê i o ô ơ u ư y
Tones	´ ` ˀ ˜ .

Fig. 1. Vietnamese written symbols

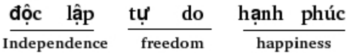


Fig. 2. Example of Vietnamese morphemes and words

For the automatic speech recognition problem, most systems for Chinese [3], Thai [4] or Vietnamese [2] share a similar approach in both acoustic modeling (AC) and language modeling (LM). Specifically, the acoustic modeling is typically based on the decomposition of a syllable into initial and final parts; while the language modeling is trained on WUs or words. As reported in [6], the performance of the system which used a word-based LM is better than the one that used a WU-based LM. However with the word-based LM approach, the search network is much bigger than the former as the vocabulary size is increased considerably. More precisely, in those systems, the standard speech recognition architecture brings all available knowledge sources very early in the process. In fact, an all-in-one search strategy is adopted which completely integrates the acoustic model with word-based LM. Subsequently, it will be more expensive in terms of memory and time, when a long-span word-based LM is exploited.

In this paper, we present a new approach for recognizing the languages mentioned above. Our approach is based on the FLaVoR architecture and exploits the compositional property of FSTs [7]. The approach consists of two steps. First, a pure acoustic-phonetic search generates a dense phoneme graph or phoneme FST, enriched with meta-data. Then, the output of the first step is composed with a series of FSTs, including sub-lexical, word lexical and word-based LM FSTs from which an usual word decoding is carried out. The word-based LM is trained by using the word segmentation procedure [6].

The paper is structured as follows. First, we briefly describe the FLaVoR architecture in Section 2. Section 3 presents our approach in detail. In Section 4, we will report the experimental result on the Vietnamese Broadcast News corpus and compare it to our previous work. Finally, some conclusions and remarks are given in Section 5.

2 The FLaVoR Approach

2.1 The Architecture

As depicted in Fig 3, the key aspect of the FLaVoR architecture consists of splitting up the search engine into two separate layers. The first layer performs

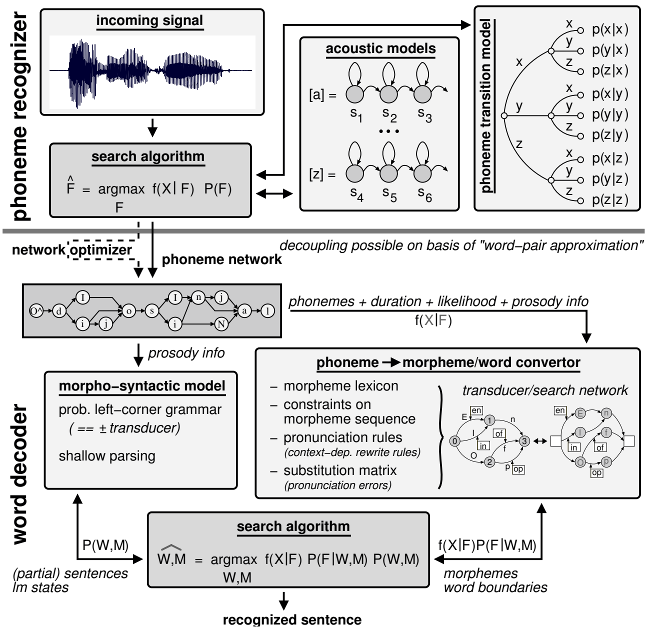


Fig. 3. FLaVoR architecture

phoneme recognition and outputs a dense phoneme network, which acts as an interface to the second layer. In this second layer, the actual word decoding is accomplished by means of sophisticated probabilistic morpho-phonological and morpho-syntactic models.

Specifically, in the first layer, a search algorithm determines the network of most probable phoneme strings F given the acoustic feature X of the incoming signal. The knowledge sources employed are an acoustic model $p(X|F)$ and a phoneme transition model $p(F)$. The isolation of the low-level acoustic-phonemic search provides the generic nature of the first layer for a full natural language. That is, the phoneme recognizer can function in any knowledge domain for a specific language.

In the word decoding stage, the search algorithm has two knowledge sources as its disposal: a morpho-phonological and a morpho-syntactic component. The morpho-phonological component converts the phoneme network into sequences of morphemes and hypothesizes word boundaries. Its knowledge sources include a morpheme lexicon, constraints on morpheme sequences, and pronunciation rules. The morpho-syntactic language model provides a probability measure for each hypothesized word based on morphological and syntactic information of the word and its context.

In this work, just a part of the FlaVoR architecture was exploited. Specifically, the PLCG, shallow parsing and searching components, as shown in Fig 3 were skipped. Instead, a full composition of transducers, including phoneme network, sub-lexion, word-based LM is performed.

2.2 Finite State Transducers

It is important to notice that all the knowledge sources mentioned above can be represented as FSTs, and that all these FSTs can be combined into one transducer. Transducers are a compact and efficient means to represents the knowledges and are a good match for the decoding process. A *Weighted finite-state transducer*, $(Q, \Sigma \cup \{\epsilon\}, \Omega \cup \{\epsilon\}, K, E, i, F, \lambda, \rho)$ is a structure with a set of states Q, an alphabet of input symbols Σ, an alphabet of output symbols Ω, a weight semiring K, a set of arcs E, a single initial state i, with weight λ and a set of final states F weighted by the function $\rho : F \rightarrow K$. A *weighted finite-state acceptor* is a weighted finite-state transducer without the output alphabet.

A composition algorithm is defined as: Let $T_1 : \Sigma^* \times \Omega^* \rightarrow K$ and $T_2 : \Omega^* \times \Gamma^* \rightarrow K$ be two transducers defined over the same semiring K. Their composition $T_1 \circ T_2$ realizes the function $T : \Sigma^* \times \Gamma^* \rightarrow K$.

Three different FST Toolkits were used for our experiments. The AT&T FSM Toolkit [7] was basically written in C, while both RWTH FSA [8] and MIT FST [9] Toolkits were written in C++, exploring the use of STL - the C++ Standard Template Library. In addition, the AT&T FSM Toolkit requires a specific license agreement in order to use its source codes, while both RWTH FSA and MIT FST are provided as open-sources. The organization of the AT&T FSM and MIT FST is similar in which each algorithm or operation is corresponding to an executable file. In contrast, the RWTH FSA combined all the algorithms and the operations into just one file. Moreover, the RWTH FSA supports labels with Unicode encoding so that it can work directly with other languages like Chinese, Japanese etc.

In the following section, we will describe how the FLaVoR architecture is applied to the Vietnamese automatic speech recognition.

3 The Proposed Approach

As suggested in Fig 4, our approach includes the following steps.

1. A phoneme transducer F is generated based on the corresponding acoustic models. F contains a set of best matching phonemes with their optimal start and end time.
2. F is composed with M, where M represents WU pronunciations, mapping phone sequences to WU sequences according to a pronouncing dictionary.
3. The resulting FST of step 2 is composed with W, where W represents word segmentations, mapping WU sequences to word sequences according to a predefined lexicon.
4. Finally, the resulting FST of step 3 is composed with word-based LM G to produces the final FST. Viterbi decoding is used to find the best path (hypothesis) through this final FST.

Thus, each path in the composition $F \circ M \circ W \circ G$ pairs a sequence of phones with a word sequence, assigning it a weight corresponding to the likelihood that

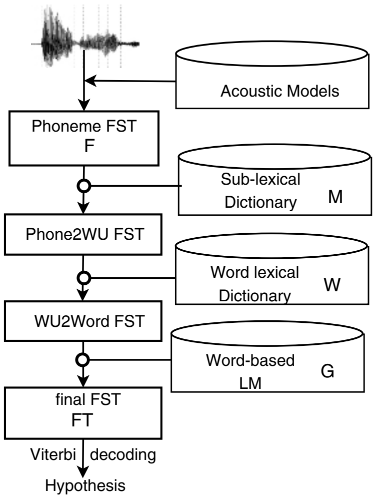


Fig. 4. The proposed approach

the word sequence is pronounced. The composition $L \circ M \circ W \circ G$ can thus serve as the modeling network for a standard Viterbi decoding in an usual way. It is important to notice that the proposed approach may not lead to a real-time system as it requires the composition and optimization of a series of FSTs.

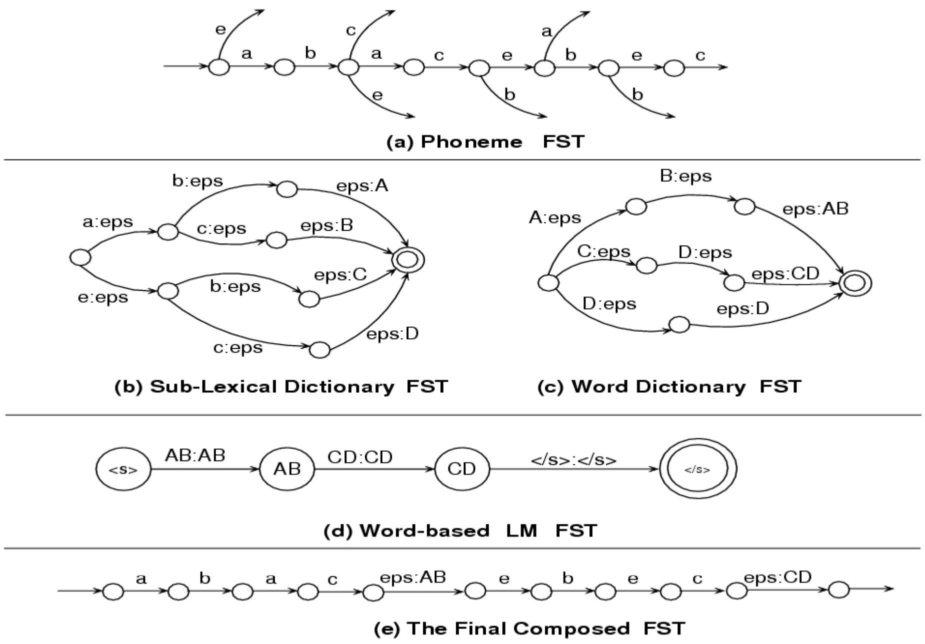
Consider the abstract example illustrated in Fig 5. In this example, there are four WUs, namely, A,B,C,D and these map respectively to four pronunciations *ab*, *ac*, *eb*, *ec*, as shown in Fig 5-b (the symbol *eps* represents the empty transition). Furthermore, there are three words, AB, CD, D, represented in the UW to word dictionary (Fig 5-c). The word-based LM is simply a bigram with its transition from AB to CD, as in Fig 5-d. Finally, the phone transducer includes a path for the phone sequence (a, b, a, c, e, b, e, c) , as given in Fig 5-a. By composing those transducers according to the procedure mentioned above, we obtain the final transducer, as depicted in Fig 5-e.

4 Experimental Results

In this section we present the experimental results of our approach on the Vietnamese Broadcast News corpus (VNBN). The results include phone error rate, word-base LM perplexity, word error rate and FST sizes.

4.1 Training Corpus and Test Data

Acoustic Training and Test Data. We used the VNBN for training and testing [2]. The acoustic training data was collected from July to August 2005 from VOV - the national radio broadcaster (mostly in Hanoi and Saigon dialects), which consists of a total of 20 hours. The recording was manually transcribed



(a) Phoneme FST

(b) Sub-Lexical Dictionary FST

(c) Word Dictionary FST

(d) Word-based LM FST

(e) The Final Composed FST

Fig. 5. An abstract example illustrating the approach

Table 1. Data for training and testing

	Training		Testing	
Dialect	Length (hours)	#Sentence	Length (hours)	#Sentence
Hanoi	18.0	17502	1.0	1021
Saigon	2.0	1994	-	
Total	20.0	19496	1.0	1021

and segmented into sentences, which resulted in a total of 19496 sentences and a vocabulary size of 3174 WUs. The corpus was further divided into two sets: training and testing, as shown in Table 1. The speech was sampled at 16kHz and 16 bits. They were further parameterized into 12 dimensional MFCC, energy, and their delta and acceleration (39 length front-end parameters).

Language Training Data. The language model training data comes from newspaper text sources. In particular, a 100M-WU collection of the national wide newspaper, VOV, was employed, which included all issues between 1998-2005 [2]. Numeric expressions and abbreviated words occurring in the texts were replaced by suitable labels. In addition, the transcriptions of the acoustic training data were also added.

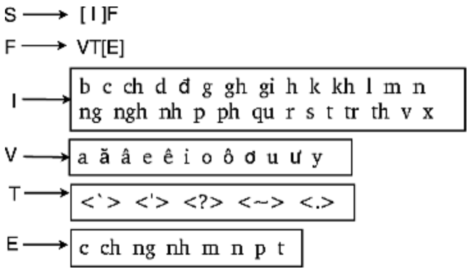
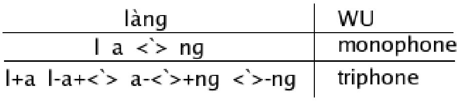

S → [I]F
F → VT[E]
I → b c ch d đ g gh gi h k kh l m n ng ngh nh p ph qu r s t tr th v x
V → a ă â e ê i o ô ơ u ư y
T → <`> <'> <?> <~> <.>
E → c ch ng nh m n p t

Fig. 6. Initial-Final Units for Vietnamese

4.2 Acoustic Models

As depicted in Fig 6, we follow the usual approach as for Chinese acoustic modeling [3] in which each syllable is decomposed into initial and final parts. While most of Vietnamese syllables consist of an initial and an final part, some of them have only the final. The initial part always corresponds to a consonant. The final part includes main sound plus tone and an optional ending sound. This decomposition results in a total number of 44 phones, as shown in Fig 6.

There is an interesting point in our decomposition scheme, which are related to a given tone in a syllable. Specifically, we treat the tone as a distinct phoneme and it follows immediately after the main sound. With this approach, the context-dependent model could be built straightforwardly. Fig 7 illustrates the process of making triphones from a syllable.



làng	WU
l a <`> ng	monophone
l+a l-a+<`> a-<`>+ng <`>-ng	triphone

Fig. 7. Construction of triphones

We use a tree-based state tying technique in which a set of 35 left and 17 right questions was designed, based on the Vietnamese linguistic knowledge. Initially, all of the states to be clustered are placed in the root node of the tree and the log likelihood of the training data calculated on the assumption that all of the states in that node are tied. This node is then split into two by finding the question which partitions the states in the parent node so as to give the maximum increase in log likelihood. The process is repeated until the likelihood increase is smaller than a predefined threshold. Fig 8 shows the split process of the decision tree for the main sound *ă*.

4.3 Language Model

Both the trigram WU-based LM and the word-based LM were trained on the text corpus mentioned above, using the SRI LM toolkit [10] with Kneser-Ney

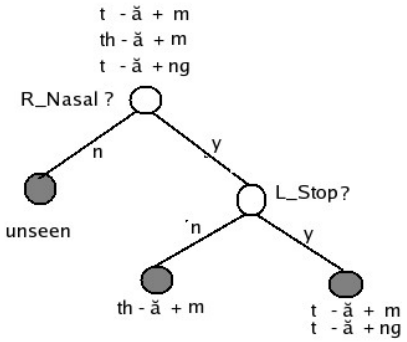


Fig. 8. Decision tree based state tying

smoothing. For the WU-based LM, a lexicon with the 5K most frequent WUs was used. This lexicon gives a 1.8% OOV rate on the newspaper corpus and about 1.0% on the VNBN. The process of building a word-based LM consists of two steps. In the first step, the WU sentences were segmented into words using the maximum match method. A Named-Entity list was also added to the original wordlist at this step to improve segmentation quality. After word segmentation, we chose the vocabulary to be the top-N most frequent words. The commonly used WUs (5K) are added then to the vocabulary as well. In particular, a lexicon consisting of 40K words and 5K WUs was selected. Table 2 reports the perplexities of both LMs on the same test set, containing 580 sentences randomly selected from VOV, issued in 2006.

Table 2. WU-based and word-based perplexities

	bigram	trigram
WU-based LM	188.6	136.2
Word-based LM	384.5	321.4

4.4 Results

Phone Recognition Results. As mentioned in [1], the key prerequisite for the proposed approach is the generation of high quality phoneme lattices in the acoustic-phonemic layer. The quality of phoneme lattices is defined by both a low phoneme error rate and a low event rate or density. The phoneme decoding was based on the architecture described in [1] in which the decoder extends the word-pair approximation to the phoneme level in order to assure a high quality of the output. Hence, to obtain high quality phoneme lattices, a phoneme transition model (an N-gram between phonemes) of a sufficiently high order N has to be used, or the LM-context has to be artificially expanded to (M-1) phonemes. The acoustic models used are context-dependent (CD) tied state phoneme models

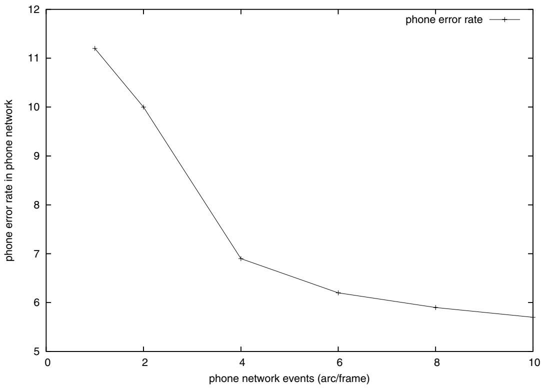


Fig. 9. Phone recognition results

(2410 states) with mixtures of tied Gaussian (48216 Gaussian) as observation density functions.

Figure 3 shows the results of phoneme recognition experiments with N=3 and M=3 and with different lattice densities. The values on the ordinate are the phoneme error rates (ins. + del. + sub.) of the phoneme lattice, i.e. the error rate of that path which matches best with the reference transcription. The values on the abscissa are the average number of events (an arc representing a CD-phoneme) per frame in the lattice. The phone graph that results in a phone error rate of 8% was selected as input to the second layer in our approach. As shown in Fig 9, this is already reached with an event rate less than 4.

Size of Transducers. Table 3 reports size of the optimized transducers, in terms of transitions. They include:

- the 5162-WU pronunciation dictionary M.
- the 39882-word lexicon W.
- the trigram word-based LM G, built as mentioned in the previous subsection.
- the phone transducer F - the report number is the average number of transitions over the sentences in the test set.
- the final composed transducer FT, also average over the test sentences.

There are two main observations obtained from the experiment. Firstly, the optimized transducers have acceptable sizes, even with a trigram word-based

Table 3. Size (number of arcs) of transducers mentioned in Fig 4

	FST(MIT)	FSA(Aachen)	FSM (AT&T)
F	4681	6112	4868
M	18741	21635	20196
W	52131	58472	54618
G	3491884	3689128	3512841
FT	58918	66121	64767

LM. Secondly, the MIT FST Toolkit performed best in the optimization, though the differences are not significant. Although the computing time is not reported here, our experiments showed that the AT&T FSM Toolkit is the fastest one.

Word Error Rate. Finally, we report the WU error rate of the new approach and compare the results with the previous work [2] (the baseline). The acoustic model in [2] is identical to the one described in this paper. The LMs however differs. In the previous work, the LM was trained on WU level while in this work, the LM was trained on word level. Moreover, both experiments used the same training and testing sets, given in Table 1.

Table 4. The WU error rate for the two approaches

	bigram	trigram
Baseline	20.8	19.1
The new approach	19.0	17.8

As shown in Table 4, the new approach shows significant improvements over the previous results. It is also observed that the WU error rate with trigram WU-based LM is roughly comparable to the one obtained with a bigram word-based LM.

5 Conclusion

We presented a different approach for recognizing the languages in the Southeast Asia, of which the boundary between words is not clear. The main advantage of our approach is that, it allows for an easier integration of different knowledge sources. In this paper, we used FST as a tool for combining the phoneme network with the word-based LM to demonstrate the idea. Experimental results on VNBN showed that our approach is both robust and flexible.

Acknowledgments

The research reported in this paper was funded by IWT in the GBOU programme, project FLaVoR (Project number 020192). `www.esat.kuleuven.be/psi/spraak/projects/FLaVoR`

References

1. Kris Demuynck, Tom Laureys, Dirk Van Compernolle and Hugo Van hamme, “FLaVoR: a Flexible Architecture for LVCSR”, Eurospeech2003, Geneva, Switzerland, pp. 1973-1976, 2003.

2. Ha Nguyen, Quan Vu,”Selection of Basic Units for Vietnamese Large Vocabulary Continuous Speech Recognition”, The 4th IEEE International Conference on Computer Science - Research, Innovation and Vision of the Future, HoChiMinh City, Vietnam, pp. 320-326, 2006.
3. Zhang J.Y. el al,”Improved Context-Dependent Acoustic Modeling for Continuous Chinese Speech Recognition”, Eurospeech2001, Aalborg, Denmark, pp 1617-1620, 2001.
4. Sinaporn Suebvisai et al, “Thai Automatic Speech Recognition”, ICASSP2005, Philadelphia, PA, USA, pp. 857- 860, 2005.
5. Liu, Y. and P. Fung, “Modeling partial pronunciation variations for spontaneous Mandarin speech recognition”, Computer Speech and Language, 17, 2003, pp. 357-379.
6. B. Xiang et al,”The BBN Mandarin Broadcast News Transcription System”, InterSpeech2005, Lisbon, Portugal, pp. 1649-1652, 2005.
7. Mehryar Mohri, Fernando C. N. Pereira, and Michael Riley. “Weighted Finite-State Transducers in Speech Recognition”, Computer Speech and Language, 16(1), pp. 69-88, 2002.
8. S. Kanthak and H. Ney, “FSA: An Efficient and Flexible C++ Toolkit for Finite State Automata Using On-Demand Computation”, ACL2004, Barcelona, Spain, pp. 510-517, 2004.
9. Lee Hetherington , “MIT Finite State Transducer Toolkit”, 2005, `http://people.csail.mit.edu/ilh//fst/`
10. Andreas Stolcke,”SRILM - An Extensible Language Modeling Toolkit”, in Proc. Intl. Conf. Spoken Language Processing, Denver, Colorado, pp. 901-904, 2002.

Language Identification by Using Syllable-Based Duration Classification on Code-Switching Speech

Dau-cheng Lyu[2,3], Ren-yuan Lyu[1], Yuang-chin Chiang[4], and Chun-nan Hsu[3]

[1] Dept. of Computer Science and Information Engineering, Chang Gung University
[2] Dept. of Electrical Engineering, Chang Gung University
[3] Institute of Information Science, Academia Sinica
[4] Institute of statistics, National Tsing Hua University
renyuan.lyu@gmail.com

Abstract. Many approaches to automatic spoken language identification (LID) on monolingual speech are successfully, but LID on the code-switching speech identifying at least 2 languages from one acoustic utterance challenges these approaches. In [6], we have successfully used one-pass approach to recognize the Chinese character on the Mandarin-Taiwanese code-switching speech. In this paper, we introduce a classification method (named syllable-based duration classification) based on three clues: recognized common tonal syllable tonal syllable, the corresponding duration and speech signal to identify specific language from code-switching speech. Experimental results show that the performance of the proposed LID approach on code-switching speech exhibits closely to that of parallel tonal syllable recognition LID system on monolingual speech.

Keywords: language identification, code-switching speech.

1 Introduction

Code-switching is defined as the use of more than one language, variety, or style by a speaker within an utterance or discourse. It is a common phenomenon in many bilingual societies. In Taiwan, at least two languages (or dialects, as some linguists prefer to call them) - Mandarin and Taiwanese- are frequently mixed and spoken in daily conversations.

For the monolingual LID system development, the parallel syllable recognition (PSR) was adopted, which is similar to the method of parallel phone recognition (PPR), and this approach is widely used in the automatic LID researches. [1,-5] Here, the reason to use syllable as the recognized result instead of phone is because both Taiwanese and Mandarin are syllabic languages. Another approach, which is called parallel phone recognition followed by language modeling (parallel PRLM), used language-dependent acoustic phone models to convert speech utterances into sequences of phone symbols with language decoding followed. After that, these acoustic and language scores are combined into language-specific scores for making an LID decision. Compared with parallel PRLM, PSR uses integrated acoustic models

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 475 – 484, 2006.

to allow the syllable recognizer to use the language-specific syllabic constraints during decoding process, and it is better than applying those constraints after syllable recognition. The most likely syllable sequence identified during recognition is optimal with respect to some combination of both the acoustics and linguistics.

However, all these approaches were confronted with an apparent difficulty. That is, they use speech signal length in sentence level or 10-45 seconds as test speech and then the language is decided by which gets maximum number of unique phonetic unit is noted as the winner for the test utterance. In our case, code-switching speech, the length of language changing may be intra-sentence or word-based level, and we can not identify the language using above approach, because there may have at least two languages embedded in a test utterance. Therefore, we have to decide the language identity in a very short time of speech utterance.

In this paper, we propose an alternative to deal with the code-switching speech LID task, which is SBDC (syllable-based duration classification). This framework could identify the language in syllable level which avoids the shortcoming of LID system in sentence or utterance-based utterances. Besides, to identify a language in each syllable that performs more precise language boundary in the code-switching speech. In this framework, we, firstly, extract acoustic and pitch features from code-switching utterance, secondly, the features are recognized as tonal syllable by our pervious recognizer [6]. Thirdly, by given the tonal syllable and its duration information, we use SBDC to identify the language for each common tonal syllable. Finally, the language smoother modifies the language identify in a statistical approach form training a code-switching speech corpus.

The structure of the paper is as follows: A LVCSR-based LID system is introduced in Section 2. The phonetic characteristic between Mandarin and Taiwanese is introduced in Section 3. In Section 4 the SBDC-based LID system is described. Finally, the performed experiments and achieved results are presented.

2 LVCSR-Based LID

It is known that LVCSR-based systems achieve high performance in language identification since they use knowledge from phoneme and phoneme sequence to word and word sequence. In [7], the LVCSR-based systems were shown to perform well in language identification. Unlike mono-lingual speech LID system [8], we implement a multi-lingual LVCSR-based system [9] as our code-switching speech LID baseline system. Fig 1 shows a block diagram of the system which includes two recognizers and each recognizer contains its won acoustic model and language model, such as AM_T, LM_T.

In this paper, the multi-lingual LVCSR-based system requires significant tonal syllable level transcribed Mandarin and Taiwanese speech data for training the acoustic and language models. During the test phase, the recognizer is employed a unified approach to recognize each tonal syllable. The step of decoding translates each tonal syllable to its won language by phonetic knowledge. It is among the most computationally complex algorithms and achieves very high language identification accuracy.

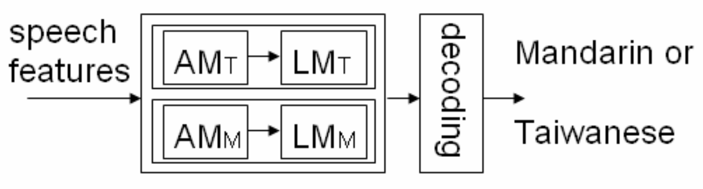


Fig. 1. A diagram of the unified LVCSR-based LID architecture for code-switching speech

3 Linguistic Characteristics of Mandarin and Taiwanese

In order to achieve reasonable identification accuracy in Taiwanese and Mandarin identification, it is necessary to understand how languages differ. They differ in the following ways:

1. Phonemic System:
2. Tone (e.g., Mandarin has four tones , Taiwanese has seven tones)
3. Tonal Syllable (there are 677 tonal syllables only belonging to Mandarin, 2231 ones only belonging to Taiwanese, and 610 tonal syllables exist in both languages by using IPA notation)
4. Lexical distribution
5. Rhythmical characteristics
6. Pace (average number of tonal syllables uttered per second) or tonal syllable duration
7. Intonation or lexical stress

The duration distribution of common tonal syllables (610) estimating from training corpus is shown in Fig 2, and they have different mean and variation. The numbers of the total sample for Mandarin are 54371 samples and 39589 samples for Taiwanese. The syllable duration of Taiwanese is about 04 sec. and 0.3 sec. for Mandarin syllable.

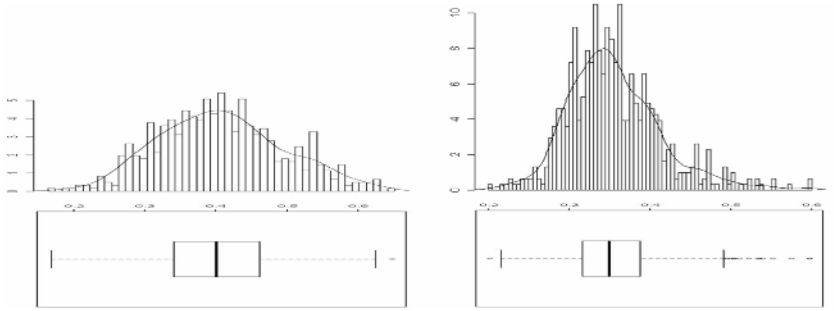


Fig. 2. The average duration distribution of common tonal syllables in Taiwanese (left) and Mandarin (right). The x-axle is the duration in second unit and y-axle is the number of common tonal syllable. (610 is the total value if the summing all y-axle value in x-axle).

4 SBDC LID

From the analysis of sec 3, we have an idea to discriminate Taiwanese from Mandarin by the duration discrepancy of the common tonal syllables. Thus, in this section, we develop a new approach to identify language for each tonal common syllable on code-switching speech.

4.1 System Overview

There are five components, including a feature extractor, a unified speech recognizer, a common tonal syllable extractor and a language smoother, in our code-switching LID system. Figure 3 illustrates the process, and the procedures are as the followings:

1) The code-switching speech input utterance is pre-extracted into a sequence of MFCCs-based feature vectors $O_T = (o_1, o_2, ..., o_T)$ with the length (frame number) *T*.
2) The unified speech recognizer [6] receives the features as the input and finds the best hypothesis tonal syllable $s^N = (s^1, s^2, ..., s^N)$ and its corresponding duration D_R, where *N* is the distinct tonal syllables for all the languages and *R* is the real number which represents the duration of each hypothesis tonal syllable. According to the pronunciation dictionary, each of the hypothesis tonal syllable is further represented by the language code $L = \{m, t, c\}$, where *m* represents Mandarin, *t* represents Taiwanese and *c* means common language. The tonal syllables with the common language mean that they exist in both Mandarin and Taiwanese, and this kind of phenomenon is caused by the union phonetic representation of the unified speech recognizer. An example is shown in the figure 4.
3) We only extract the speech segment with common language, S_c, for discriminating between Mandarin and Taiwanese.
4) The three parameters, O_T, S_c and D_R are as the inputs to train the syllable-based duration classifier (SBDC). The output is language specific tonal syllable, S_{ct} and S_{cm} for instance. This part will describe particularly in the section 4.2.
5) In practice, the unit of code-switching language appears as a word whose unit exceeds duration. Under this assumption, the smoothing process is involved to eliminate the unreasonable language switching with a short interval by the language modeling after joining the parts of S_{cm}, S_{ct} and S_t, S_m. The final output is $S_{\tilde{m}}$ or $S_{\tilde{t}}$ which is a tonal syllable with the language identity of Mandarin or Taiwanese.

4.2 Probabilistic Framework

The most likely language L_i by given three parameters: the acoustic information O_T, the common tonal syllable S_c, and its duration D_R, is found using the following expression:

$$Li(O_T) = \arg\max_i P(L_i \mid O_T, S_c, D_R) \tag{1}$$

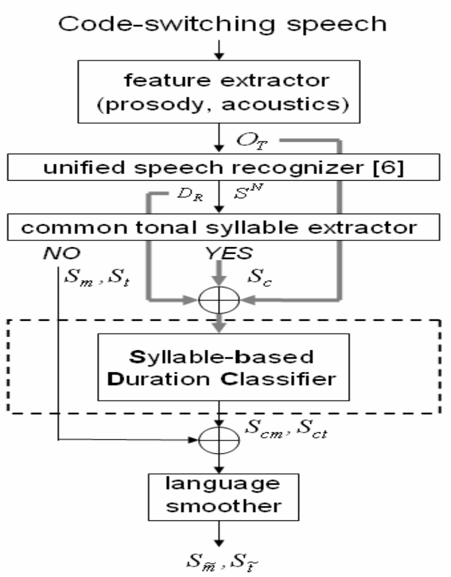

Code-switching speech
feature extractor (prosody, acoustics)
O_T
unified speech recognizer [6]
D_R
S^N
common tonal syllable extractor
NO
YES
S_m, S_t
S_c
Syllable-based Duration Classifier
S_{cm}, S_{ct}
language smoother
$S_{\tilde{m}}, S_{\tilde{t}}$

Fig. 3. The flow chart of SBDC LID system

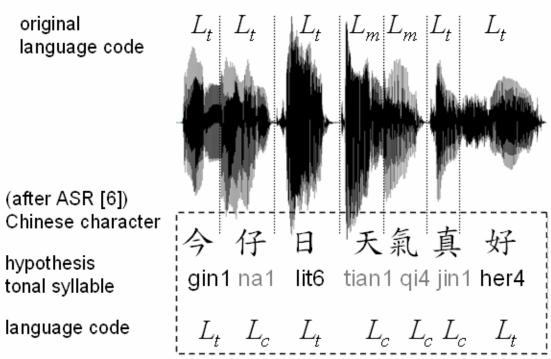

original language code
L_t L_t L_t L_m L_m L_t L_t
(after ASR [6]) Chinese character
今 仔 日 天 氣 真 好
hypothesis tonal syllable
gin1 na1 lit6 tian1 qi4 jin1 her4
language code
L_t L_c L_t L_c L_c L_c L_t

Fig. 4. Example of language code

Using standard probability theory, this expression can be equivalently written as

$$L_i(O_T) = \underset{i}{\operatorname{argmax}} P(O_T \mid L_i, S_c, D_R) P(D_R \mid L_i, S_c) P(S_c \mid L_i) P(L_i) \qquad (2)$$

The four probability expressions in (2) are organized in such a way that duration and common tonal syllable information are contained in separate terms. In modeling, these terms become known as

1. $P(O_T \mid L_i, S_c, D_R)$ Common tonal syllable acoustic model.
2. $P(D_R \mid L_i, S_c)$ Duration model.

3. $P(S_c \mid L_i)$ The phonetic language model.
4. $P(L_i)$ The a priori language probability.

Assuming that a priori language probability for each language on code-switching speech is equal, and phonetic language model for common hypothesis tonal syllable is also equal. The hypothesized language is determined by maximizing the log-likelihood of language L_i with respect to speech O_T and is estimated as follows:

$$L_i(O_T) = \underset{i}{\arg\max}\{\log P(O_T \mid L_i, S_c, D_R) + \log P(D_R \mid L_i, S_c)\} \tag{3}$$

According to [3], the syllabic information is contained in two separate models: the syllabic acoustic model and the syllabic duration model, which are shown in Fig 5. In subsequent sections these models will simply be referred to as the acoustic model and the duration model. The acoustic model accounts for the different acoustic realizations of the syllabic elements that may occur across languages, whereas the duration model accounts for the probability distributions of the syllabic elements, and captures the differences that can occur in duration structures of different languages due to the boundary or segmented created by variations in the common tonal syllabic durations. This organization provides a useful structure for evaluating the relative contribution towards language identification that acoustic and duration information provide.

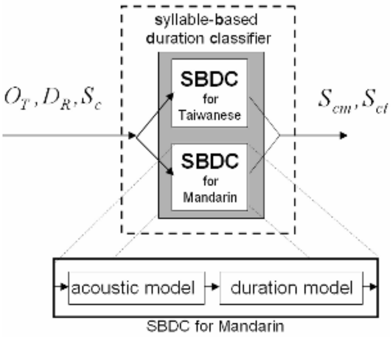


Fig. 5. Illustration of syllable-based duration classifier component

4.3 Acoustic Model of SBDC

The expression $P(O_T \mid S_c, D_R, L_i)$ is called the acoustic model, which is used to capture information about the acoustic realizations of each of the common tonal syllable used in each language. However, the duration of each common tonal syllable is a real number form 0 to 1 and that is parametric difficultly. To simplify the parameter of the acoustic model, like the idea of [1], the duration *R* of the D_R for each common tonal syllable S_c is quantized into two levels: long and short, by the following steps:

Step1: Forced alignment:
In the training phase, we need to get the duration for each tonal syllable, because our transcription of the training speech only contains the pronunciation, not including

duration information. Therefore, we used HMM-based method to get the duration value of each tonal syllable by forced alignment approach on training corpora for both languages.

Step2: Average duration estimation:

A histogram of duration for each tonal syllable emitted from forced alignment is collected and the average duration determined.

Step3: Quantization:

A –L suffix is appended to all tonal syllables having duration longer than the average duration for that tonal syllable, and –S suffix is appended to all tonal syllables having duration shorter than the average duration for that tonal syllable. The diagram for these steps for language *i* is illustrated in the Fig. 6

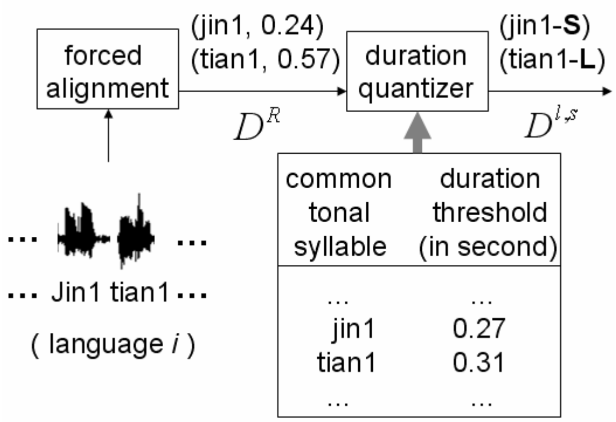


Fig. 6. An example of tagging quantized duration for each common tonal syllable (Form D_R to D_l or D_s)

4.4 Duration Model of SBDC

The expression $P(D_R \mid L_i, S_c)$ captures the segment duration information in a common tonal syllable. While there may be very useful information to separate the difference between Mandarin and Taiwanese in syllable level. The probability can be modeled with a mixture of Gaussian models. The Gaussians in each mixture are then iteratively re-estimated to maximize the average likelihood score of the vectors in the training data. To ensure proper amounts of training data for each mixture of Gaussians, the number of Gaussian used to model each syllable in each language is determined by the amount of the training data.

4.5 Language Smoother (LS)

The goal of the language smoother is to modify the language identity to be more reasonable in language switching by an N-gram language model trained from a real code-switching corpus. An example is shown in Fig 7.

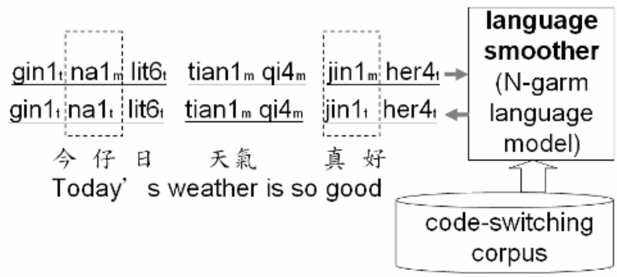

gin1t na1m lit6t tian1m qi4m jin1m her4t
gin1t na1t lit6t tian1m qi4m jin1t her4t
今 仔 日 天氣 真 好
Today' s weather is so good
language smoother (N-garm language model)
code-switching corpus

Fig. 7. An example for merging language identification results by language smoother

5 Experiments and Results

The goal of the experiment is to verify that SBDC-based system could have high accuracy syllable LID rate and to outperform a LVCSR-based system. In addition, we also evaluate the performance of our proposed approach is close to that on monolingual speech which maybe the upper bound performance on code-switching speech.

5.1 Corpus and Experiment Setup

The speech corpus used in all the experiments were divided into three parts, namely, the training set, evaluating set and the testing set. The training set consists of two mono-lingual Taiwanese and Mandarin speech data, which includes 100 speakers. Each speaker read about 700 phonetically abundant utterances in both languages. The evaluating set is to train the back-off bi-gram code-switching language model, which estimates the probability of language translation, and adapts the threshold of syllable duration quantizer. For testing data set, another 12 speakers were asked to record 3000 Mandarin-Taiwanese code-switching utterances. Among these utterances, at least one Taiwanese word is embedded into a Mandarin carrier sentence. The length of each word is various from one to eight syllables. The statistics of the corpus used here are listed in Table 1.

The acoustic features used in SBDC are the same with in [5], they are: mel-frequency cepstral coefficients (MFCC) which includes 12 cepstral coefficients, normalized energy and prosody information. The first and second derivatives of parameters are also included. The acoustic model of SBDC is used HMM-based approach in tonal syllable unit with duration tag for both languages, and each HMM has seven states. We have 2440 (610 common tonal syllables, and each one has two duration classes and two languages) HMM, and the final number of the mixture in each state dependents on the occurrence of training data. The more the training data of the state has, the more of the mixture number is.

5.2 Results

We compare the LID performance on two different types of speech: f monolingual speech, and on code-switching speech. The contents of these two sets are the same, because we used manual segment to extract the part of monolingual speech form code-switching speech. For the monolingual speech, we used parallel tonal syllable

Table 1. Statistics of the bi-lingual speech corpus used for training and testing sets. M: Mandarin, T: Taiwanese, CS: code-switching utterances.

	Language	No. of Speakers	No. of Syllable	No. of Hours
Training set	M	100	112,032	11.3
	T	100	124,768	11.2
Evaluating set	CS	4	12,680	1.10
Test set	CS.	12	41,681	3.31

recognition approach [5], and the approach is similar with PPR [1]. On the other hands, for the code-switching speech, we used three approaches which are LVCSR-based approach, SBDC and SBDC+LS. The last one is SBDC approach adding the language smoother. The 10K and 20K vocabulary size of SBDC and SBDC+LS approaches are followed the experimental condition of [6], because, before do SBDC approach, we need the recognized tonal syllable form unified ASR in [6]. The results are listed in the Table2.

The experimental result for the monolingual speech has the highest LID accuracy rate, 88.05%, and this is the experimental upper bound performance of LID on code-switching speech using our method.

On the code-switching speech, using SBDC+LS approach has the best performance which is close to that in the monolingual speech. Both of the performances of using SBDC approach are more outstanding than LVCSR-based approach, the critical reason is according to the useful information such as duration of tonal syllable used during decision process.

Table 2. The LID accuracy rate for different approaches

	LID accuracy rate (%)	
monolingual speech	**88.05**	
code-switching speech	10K	20K
LVCSR-based	82.14	81.93
SBDC	86.08	84.78
SBDC+LS	**87.53**	**85.91**

6 Conclusion

In this paper, we used three clues: recognized common tonal syllable tonal syllable, the corresponding duration and speech signal, building a SBDC LID system to identify specific language from code-switching speech. The system's architecture is factorized as HMM-based quantized duration acoustic model in tonal syllable, GMM-based duration model and language smoother. The experimental results show a promising performance on LID accuracy rate to compare with the LVCSR-based system and the performance also approaches that in monolingual speech by using PSR method.

References

1. Zissman, M. A. "Comparison of four Applications to Automatic Language Identification of Telephone Speech," IEEE Trans. on Speech and Audio Proc., Vol. 4, No. 1, pp. 31-44, 1996
2. T. Nagarajan and Hema A. Murthy, "Language Identification Using Parallel Syllable-Like Unit Recognition," ICASSP, 2004
3. Hazen, T. J., & Zue, V. W., "Segment-Based Automatic Language Identification," Journal of Acoustic Society of America, April 1997.
4. Rongqing Huang, John H.L. Hansen, "DIALECT/ACCENT CLASSIFICATION VIA BOOSTED WORD MODELING," ICASSP 2005
5. Pedro A. Torres-Carrasquillo, Douglas A. Reynolds, J. R. Deller, Jr., "Language Identification Using Gaussian Mixture Model Tokenization," Proc. ICASSP 2002, pp. I-757-760.
6. Dau-Cheng Lyu, Ren-Yuan Lyu, Yuang-chin Chiang and Chun-Nan Hsu, "Speech Recognition on Code-Switching Among the Chinese Dialects," ICASSP, 2006
7. T. Schultz et al., "LVCSR-based Language Identification," Proc. ICASSP, pp. 781-784, Altlanta 1996.
8. J.L.Hieronymus, S.Kadambe, "Robust Spoken Language Identification using Large Vocabulary Speech Recognition", ICASSP, vol.2, pp.1111-1114, Munich, Germany, Apr., 1997
9. Santhosh C. Kumar, VP Mohandas and Haizhou Li, "Multilingual Speech Recognition: A Unified Approach", InterSpeech 2005 - Eurospeech - 9th European Conference on Speech Communication and Technology, September 4-8, 2005, Lisboa, Portugal

CCC Speaker Recognition Evaluation 2006: Overview, Methods, Data, Results and Perspective

Thomas Fang Zheng[1,2], Zhanjiang Song[2,3], Lihong Zhang[3], Michael Brasser[1,3],
Wei Wu[2], and Jing Deng[2]

[1] Chinese Corpus Consortium (CCC)
fzheng@tsinghua.edu.cn, mbrasser@d-Ear.com
http://www.CCCForum.org
[2] Center for Speech Technology, Tsinghua National Laboratory for
Information Science and Technology, Tsinghua University, Beijing, 100084
{fzheng, szj, wuwei, dengj}@cst.cs.tsinghue.edu.cn
[3] Beijing d-Ear Technologies Co., Ltd.
{zjsong, lhzhang, mbrasser}@d-Ear.com
http://www.d-Ear.com

Abstract. For the special session on speaker recognition of the *5th International Symposium on Chinese Spoken Language Processing* (ISCSLP 2006), the *Chinese Corpus Consortium* (CCC), the session organizer, developed a speaker recognition evaluation (SRE) to act as a platform for developers in this field to evaluate their speaker recognition systems using two databases provided by the CCC. In this paper, the objective of the evaluation, and the methods and the data used are described. The results of the evaluation are also presented.

Keywords: Speaker recognition, Evaluation.

1 Introduction

Speaker recognition (or voiceprint recognition, VPR) is an important branch of speech processing with applications in many fields, including public security, anti-terrorism, forensics, telephony banking, and personal services. However, there are still many fundamental and theoretical problems to solve, such as issues with background noise, cross-channel recognition, multi-speaker recognition, and difficulties arising from short speech segments for training and testing [1-3].

In addition to inviting researchers to present their state-of-the-art achievements in various aspects of the speaker recognition field, this special session on speaker recognition of the *5th International Symposium on Chinese Spoken Language Processing* (ISCSLP 2006) provides a platform for VPR developers to evaluate their speaker recognition systems using two databases provided by the *Chinese Corpus Consortium* (CCC). This paper is organized as follows. In Section 2, an overview of

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 485 – 493, 2006.

the evaluation is given. Details of the evaluation are described in Section 3. The summary and further perspectives on the evaluation are given in Section 4.

2 Overview of the Evaluation

2.1 Organizer

This speaker recognition evaluation (SRE) was organized by the CCC. The CCC was founded in March 2004, sponsored by Dr. Thomas Fang Zheng and co-founded by 8 universities, institutes and companies. The aim of the CCC is to provide corpora for Chinese ASR, TTS, NLP, perception analysis, phonetics analysis, linguistic analysis, and other related tasks. The corpora can be speech- or text-based; read or spontaneous; wideband or narrowband; standard or dialectal Chinese; clean or with noise; or of any other kinds which are deemed helpful for the aforementioned purposes. Currently there are numerous corpora available from the CCC. For more information, readers can refer to the official website of the CCC (http://www.CCCForum.org) and paper [4].

2.2 Objective

The purpose of this SRE is to provide an opportunity for VPR researchers and developers to exchange their ideas and to help push forward, especially, corresponding work on Chinese language data. It can be seen as a specially focused event, similar to other well-known events (e.g. the speaker recognition evaluations carried out by NIST [5-7]).

3 The CCC 2006 SRE

Detailed information on the CCC 2006 SRE is given in this section.

3.1 Task Definition

The CCC 2006 SRE covers the following six tasks:

1) Text-dependent single-channel speaker verification.
2) Text-independent single-channel speaker verification.
3) Text-dependent cross-channel speaker verification.
4) Text-independent cross-channel speaker verification.
5) Text-independent single-channel speaker identification.
6) Text-independent cross-channel speaker identification.

All of the above tasks are optional for participants.

Please note that for text-dependent speaker-verification tasks in this evaluation (both single-channel and cross-channel), a test sample is treated as a true speaker trial only when both the speaker identity and the content match those of the training samples.

3.2 Performance Measure

The methods for measuring the performance of the participating systems are described below.
(1) Speaker Verification

The performance of a speaker verification system is evaluated by a Detection Error Tradeoff (DET) curve and a detection cost function (C_{Det}) [6]. The $C_{Det}$is defined as a weighted sum of miss and false alarm error probabilities:

$$C_{Det} = C_{Miss} \times P_{Miss} \times P_{Target} + C_{FalseAlarm} \times P_{FalseAlarm} \times \left(1 - P_{Target}\right) \tag{1}$$

where C_{Miss} and $C_{FalseAlarm}$ are the relative costs of miss errors and false alarm errors, and P_{Target} is the *a priori* probability of the specified target speaker (in this evaluation, these parameters are set as in Table 1). P_{Miss} and $P_{FalseAlarm}$ are the miss probability and false-alarm probability, respectively. A miss error occurs when a true speaker model of a test segment is rejected, while a false alarm error occurs when an impostor model of a test segment is accepted. The miss probability is defined as

$$P_{Miss} = \frac{N_{Miss}}{N_{VS}} \times 100\% \tag{2}$$

where N_{Miss} is the number of miss errors and N_{VS} is the number of true speaker trials. The false alarm probability is defined as

$$P_{FalseAlarm} = \frac{N_{FalseAlarm}}{N_{VI}} \times 100\% \tag{3}$$

where $N_{FalseAlarm}$ is the number of false alarm errors and N_{VI} is the number of impostor trials.

Table 1. Speaker verification cost model parameters

C_{Miss}	$C_{FalseAlarm}$	P_{Target}
10	1	0.05

(2) Speaker Identification

The performance of a speaker identification system is evaluated by its *Identification Correctness Rate* (P_{IC}), which is defined as:

$$P_{IC} = \frac{N_{IC}}{N_{IT}} \times 100\% \tag{4}$$

where N_{IC} is the number of correctly identified segments. A correctly identified segment means that the system should output the model speaker's identity as top-candidate for "in-set" tests, and output a "non-match" flag for "out-of-set" tests. N_{IT} is the total number of trial segments.

3.3 Corpora

The data sets, including development data and evaluation data, were extracted from two CCC databases, CCC-VPR3C2005 and CCC-VPR2C2005-1000.

CCC-VPR3C2005: This corpus contains two subsets, one for text-independent VPR and the other for text-dependent VPR. This corpus can also be used for multi-channel or cross-channel VPR research, because each sentence (in Chinese) was simultaneously recorded through three different types of microphones. The three types of microphones are labeled with 'U', 'L', and 'R', respectively. All samples are stored in Microsoft wave format files with a 48 kHz sampling rate, 16-bit PCM, and mono-channel.

CCC-VPR2C2005-1000: This corpus contains speech from 1,000 male speakers aged 18-23, each of whom was required to utter 40 Chinese sentences in the given order. All utterances were required to be made twice, speaking clearly and naturally without any attempt to disguise the voice. For each speaker, the first time the utterance was recorded through a GSM mobile phone and the second time the utterance was recorded through a landline telephone.

For more details on these two data sets, please visit the homepage of the CCC and check their corresponding links on the "*Corpora*" page.

Although the participants were allowed to use the data set(s) they already had to develop their system(s), the CCC also provided them with development data, and all tests were performed on the evaluation data later provided by the CCC. All the wave files in the selected data sets are of 8 kHz sample rate, 16-bit precision, mono, linear PCM format (some of them were converted from different sample rates).

3.4 Development Data

Two development data sets were provided, one for text-independent tasks and one for text-dependent tasks.

(1) Development Data for Text-Independent Tasks

This data set is taken from CCC-VPR2C2005-1000. It contains data from 300 speakers randomly selected from the original 1,000 speakers. Data for each speaker includes 2 utterances, corresponding to one land-line (PSTN) channel utterance and one cellular-phone (GSM only) channel utterance. So the development data includes a total of 600 (=300×2) utterances.

Each utterance is divided into several segments, where there is at least 1 segment longer than 30 seconds, which can be used to train the speaker model. The other part is divided into several shorter segments, which can be used for testing. The order of the segments of different lengths in an utterance is determined randomly.

The relationships between the segment files and their speaker identities are defined in a key file shipped with the data set. This file also includes other necessary information, such as channel type and gender.

(2) Development Data for Text-Dependent Tasks

This data set is taken from CCC-VPR3C2005. It contains utterances partly selected from 5 male speakers' data and 5 female speakers' data. The data can be used as samples to listen or to perform some simple tests, but it is not sufficient to be used for clustering, for example, training channel-specific UBMs as the other data set can. In this data set, each speaker's data comes from three microphones, marked as micl, micr, and micu, respectively. For each channel, the data for each speaker includes 5 utterances repeated 4 times, as well as 21 other unrepeated utterances. The relationship between the segment files and their speaker identity are defined in a key file shipped with the data set. This file also includes other necessary information, including channel type and gender.

This data set also provides transcriptions for the training utterances, which can be accessed via the indexes listed in the key file.

3.5 Evaluation Data

The general features of the evaluation data, such as involved channel types and speaking styles, are the same as those of the development data. However, the speakers in the two stages' data sets do not overlap.

The training and trial lists were shipped with the evaluation data set, which covers the predefined evaluation tasks, i.e., combinations of text-independent or text-dependent, identification or verification, single-channel or cross-channel. For verification tasks, the ratio of testing samples for true-speakers and imposters is about 1:20; while for identification tasks, the ratio of testing samples for in-set (matched) and out-of-set (non-matched) cases is about 1:1.

The key files mapping test samples with their speaker identities were sent to the participants, along with the performance rankings and evaluation scripts, after all results were received and verified.

3.6 Participants

Eight research sites participated in the CCC 2006 SRE. The sites and their affiliations are:

- **NTUT-EE:** Speech Lab, Department of Electronic Engineering, National Taipei University of Technology, Taipei.
- **UVA-CS:** Computer Science Department, Universidad de Valladolid, Valladolid
- **CUHK-EE:** Department of Electronic Engineering, The Chinese University of Hong Kong, HKSAR.
- **THU-EE:** Department of Electronic Engineering, Tsinghua University, Beijing.

- **I2R-SDPG:** Speech and Dialogue Processing Group, Institute for Infocomm Research, Singapore.
- **EPITA:** BiOSECURE-EPiTA-FRiBOURG-GET, Le KREMLiN-BiCETRE
- **UT-ITS:** Institute of Technology and Science, The University of Tokushima, Tokushima
- **SINICA-IIS:** Institute of Information Science, Academia Sinica, Taipei.

3.7 Results

Although in total there were 6 tasks, no results for text-dependent single-channel speaker verification were submitted. A total of 17 systems from the eight participants were submitted for the remaining 5 tasks.

(1) Identification tasks

Only one test result for the text-independent cross-channel speaker identification task (abbreviated as *i-ti-c*) and two test results for the text-independent single-channel speaker identification task (abbreviated as *i-ti-s*) were submitted. The P_{IC}'s of these systems are shown in Table 2.

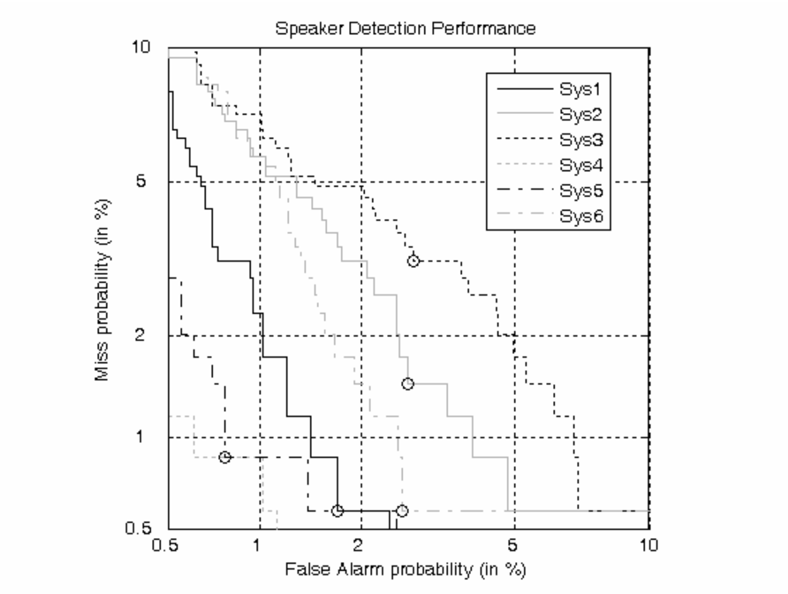


Fig. 1. DET curves for the *v-ti-s* task

(2) Verification tasks

The remaining 14 systems were for the verification tasks, particularly, 6 for the text-independent single-channel speaker verification task (abbreviated as *v-ti-s*), 7 for the text-independent cross-channel speaker verification task (abbreviated as *v-ti-c*) and 1 for the text-dependent cross-channel speaker verification task (abbreviated as *v-td-c*).

Table 2. Identification test results

	i-ti-c	*i-ti-s*
Sys1	86.45%	
Sys2		99.33%
Sys3		97.16%

The DET curves and corresponding minimum C_{Det}s for the above tasks are given in Fig. 1 and Table 3, Fig. 2 and Table 4, Fig. 3 and Table 5, respectively. Note that the system IDs for each task are assigned independently.

Table 3. The minimum C_{Det}s for the systems in the *v-ti-s* task

System	Sys1	Sys2	Sys3	Sys4	Sys5	Sys6
C_{Det} (×100)	1.1	2.1	3.0	0.6	0.8	1.6

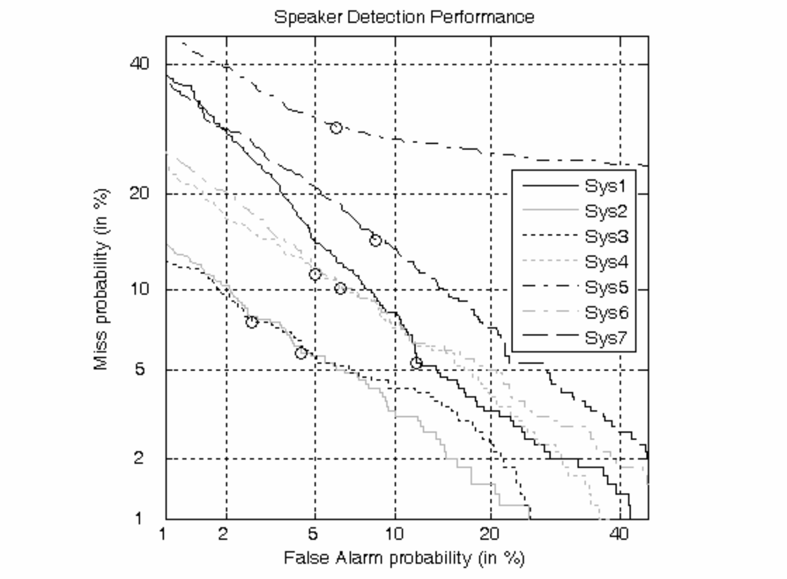


Fig. 2. DET curves for the *v-ti-c* task

Table 4. The minimum C_{Det}s for the systems in the *v-ti-c* task

System	Sys1	Sys2	Sys3	Sys4	Sys5	Sys6	Sys7
C_{Det} (×100)	8.6	5.1	5.2	8.2	17.8	8.2	11.5

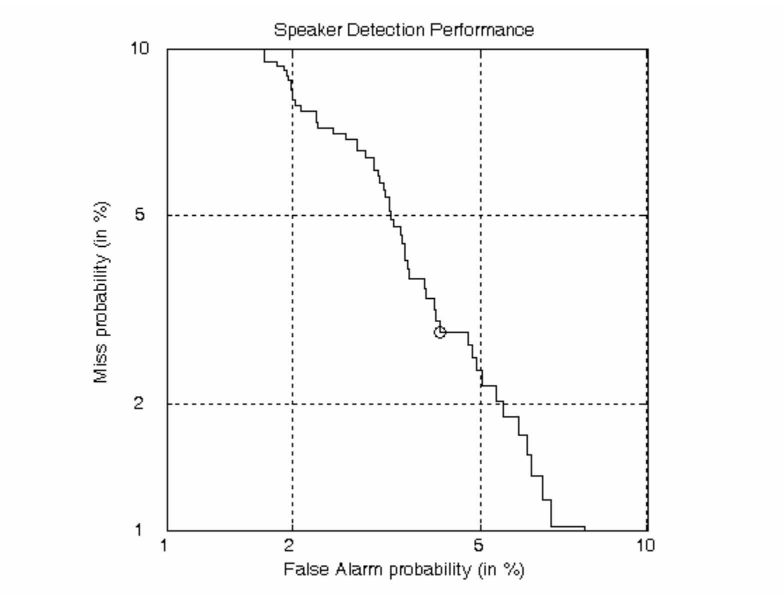


Fig. 3. DET curves for the *v-td-c* task

Table 5. The minimum C_{Det}s for the systems in the *v-td-c* task

System	Sys1
C_{Det} (×100)	3.53

As shown in the above results for the text-independent identification and verification tasks, the overall system performance in a cross-channel environment is worse than that in a single-channel environment, even though the cross-channel environment involves only two channel types, GSM and land-line. This phenomenon reveals that the channel effect is still a great impediment for speaker recognition. In light of this, the CCC is planning to collect corpora covering more complicated cross-channel environments, including various transmission channels and handsets.

4 Summary and Perspective

The CCC 2006 SRE began on Feb. 01, 2006 [8], and the conference presentation will be held on Dec. 16, 2006. Although this is the first time for this evaluation event to be carried out, the CCC would like to continuously support, improve and develop it into a series of events in the near future. This SRE was designed to be open to all, with announced schedules, written evaluation plans and follow-up workshops. The purpose of the evaluation is to provide additional chances for researchers and developers in this field to exchange their ideas and to help push forward, especially, corresponding work on Chinese language data. The CCC intends to use the experience gained this year in designing future evaluations. Any site or research group desiring to participate in future evaluations is welcome, and should contact Dr. Thomas Fang Zheng (fzheng@tsinghua.edu.cn).

References

1. Campbell, J. P.: Speaker recognition: A tutorial. Proceedings of the IEEE. 85(9):1437-1462 (1997)
2. Reynolds, D. A.: An overview of automatic speaker recognition technology. In Proc. of ICASSP, 5: 4072-4075, Orlando, Florida, (2002)
3. Bimbot F., Bonastre J.-F., Fredouille C., Gravier G., Magrin-Chagnolleau I., Meignier S., Merlin T., Ortega-Garcia J., Petrovska-Delacretaz D., and Reynolds D.-A.: A tutorial on text-independent speaker verification. EURASIP Journal on Applied Signal Processing, 4:430–451 (2004)
4. Zheng, T. F.: The Voiceprint Recognition Activities over China-Standardization and Resources. Oriental COCOSDA 2005, pp.54-58, December 6-8, Jakarta, Indonesia (2005)
5. Przybocki, M. A. and Martin, A. F.: NIST speaker recognition evaluations. In Proceedings of the International Conference on Language Resources and Evaluation (LREC), pp. 331-335, Grenada, Spain, (1998).
6. Doddington, G.R., Przybycki, M.A., Martin, A.F., Reynolds, D.A.: The NIST speaker recognition evaluation - Overview, methodology, systems, results, perspective. Speech Commun. 31(2-3) (2000) pp.225-254.
7. Martin, A. and Przybocki, M.: The NIST Speaker Recognition Evaluations: 1996-2001. In Proc. 2001: A. Speaker Odyssey, The Speaker Recognition Workshop, Crete, Greece, June 18-22, 2001, pp. 39-43.
8. http://www.iscslp2006.org/specialsessions.htm

The IIR Submission to CSLP 2006 Speaker Recognition Evaluation

Kong-Aik Lee[1], Hanwu Sun[1], Rong Tong[1], Bin Ma[1], Minghui Dong[1],
Changhuai You[1], Donglai Zhu[1], Chin-Wei Eugene Koh[2], Lei Wang[2],
Tomi Kinnunen[1], Eng-Siong Chng[2], and Haizhou Li[1,2]

[1] Institute for Infocomm Research,
21 Heng Mui Keng Terrace, Singapore 119613
kalee@i2r.a-star.edu.sg
[2] School of Computer Engineering,
Nanyang Technological University, Singapore 639798
{aseschng, hzli}@ntu.edu.sg

Abstract. This paper describes the design and implementation of a practical automatic speaker recognition system for the CSLP speaker recognition evaluation (SRE). The speaker recognition system is built upon four subsystems using speaker information from acoustic spectral features. In addition to the conventional spectral features, a novel *temporal discrete cosine transform* (TDCT) feature is introduced in order to capture long-term speech dynamic. The speaker information is modeled using two complementary speaker modeling techniques, namely, Gaussian mixture model (GMM) and support vector machine (SVM). The resulting subsystems are then integrated at the score level through a multilayer perceptron (MLP) neural network. Evaluation results confirm that the feature selection, classifier design, and fusion strategy are successful, giving rise to an effective speaker recognition system.

1 Introduction

Speaker recognition is the process of automatically establishing personal identity information by analyzing speech utterances [1]. The goal of speaker recognition is to identify people by voice. This paper describes and evaluates an automatic speaker recognition system that addresses two different tasks, namely, speaker verification and speaker identification. Speaker verification is the task of validating a claimed identity, whereas speaker identification refers to the task of determining who is speaking [1, 2]. Speaker recognition technology has been found important in various applications, such as, public security, anti-terrorism, justice, and telephone banking.

As part of the 5th *International Symposium on Chinese Spoken Language Processing* (ISCSLP 2006), a special session on speaker recognition is organized by the *Chinese Corpus Consortium* (CCC) [3]. The CSLP speaker recognition evaluation (SRE) aims to provide a common platform for researchers to evaluate their speaker recognition systems. The focus of the CSLP SRE is on Chinese speech, as opposed to some other well-known SRE events, e.g., those carried out by *National Institute of Standards and Technology* (NIST) [4], which focus on English speech. The CSLP

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 494 – 505, 2006.

2006 SRE includes text-dependent and text-independent speaker recognition tasks under single-channel and cross-channel training-testing conditions. In this paper we focus on the text-independent speaker verification and identification tasks.

The development and evaluation sets provided for the text-independent tasks of the CSLP 2006 SRE are derived from the CCC-VPR2C2005-1000 corpus (*CCC 2-channel corpus for voiceprint recognition 2005–1000 speakers*) [3]. The development set contains telephone speech utterances from 300 male speakers, while the evaluation set involves 700 male speakers. The speakers in the two datasets do not overlap. In both datasets, the duration of training samples is guaranteed to be approximately longer than 30 seconds, however, the test segments are much shorter.

Table 1. CSLP 2006 SRE evaluation categories

		Text independent	Text-dependent
Speaker verification	Single channel	×	▨
	Cross channel	×	▨
Speaker identification	Single channel	×	
	Cross channel	×	

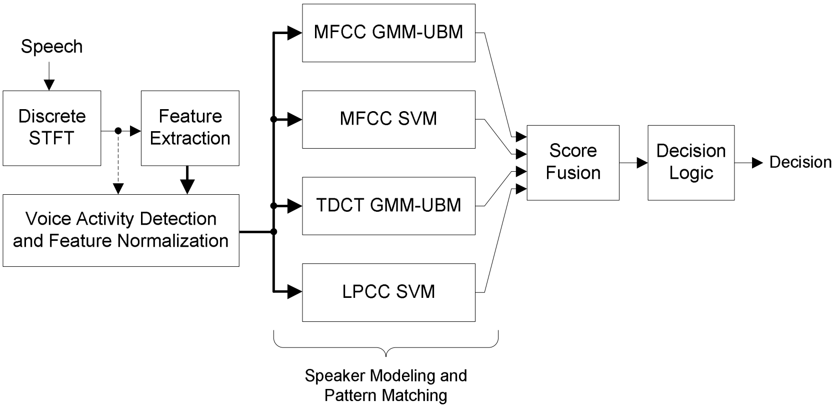


Fig. 1. An automatic speaker recognition system built upon four subsystems. Three different features (MFCC, LPCC, and TDCT) and two different speaker modeling techniques (SVM and GMM) are employed in the subsystems.

This paper describes the design and implementation of a practical automatic speaker recognition system for the CSLP 2006 SRE. The *Speech and Dialogue Processing Group* of *Institute for Infocomm Research* (IIR) participates in four (see checked boxes in Table 1) out of the six evaluation categories (see shaded boxes in Table 1) of this year SRE event. Our submission is built upon four subsystems using speaker information from acoustic spectral features [2, 5, 6, 7], as illustrated in Fig. 1. The speaker information represented in various forms is modeled using Gaussian mixture model (GMM) [7, 8] and support vector machine (SVM) [9, 10]. Feature extraction and speaker modeling techniques employed in the subsystems are described in Section 2 and Section 3, respectively. The specifications of the subsystems, together with the system integration issue, are then detailed in Section 4. In Section 5, the evaluation results are presented. Finally, Section 6 concludes the paper.

2 Feature Extraction

As the front-end of the automatic speaker recognition system, the function of the feature extraction is to parameterize an input speech signal into a sequence of feature vectors [2]. The purpose of such transformation is to obtain a new representation of the speech signal, which is more compact and allows a tractable statistical modeling. Our speaker recognition system uses two basic sets of acoustic spectral features, namely, the *mel-frequency cepstral coefficients* (MFCC) and the *linear prediction cepstral coefficients* (LPCC) [2, 5, 7]. A third set of features is derived from the MFCC features by taking the discrete cosine transform (DCT) along the time axis, hence the name *temporal DCT* (TDCT) features [6].

2.1 Mel-Frequency Cepstral Coefficients

Prior to feature extraction, the input speech signal is pre-emphasized using a first order finite impulse response filter (FIR) with its zero located at $z = 0.97$. The pre-emphasis filter enhances the high frequencies of the speech signal, which are generally reduced by the speech production process [7].

MFCC feature extraction begins by applying a discrete short-time Fourier transform (STFT) on the pre-emphasized speech signal, using a 30 ms Hamming window with 10 ms overlap between frames. The magnitude spectrum of each speech frame, in the frequency range of 0 to 4000 Hz, is then weighted by a set of 27 mel-scale filters [5]. The mel-scale filter bank emulates the critical band filters of human hearing mechanism. Finally, a 27-point DCT is applied on the log energy of the mel-scale filter bank outputs giving rise to 27 cepstral coefficients. The first coefficient is discarded, and the subsequent 12 coefficients are taken to form a cepstral vector. Delta and delta-delta features are computed over a ± 1 frame span and appended to the cepstral vector, forming a 36-dimensional MFCC feature vector. The delta and delta-delta features contain the dynamic information about the way the cepstral features vary in time.

2.2 Linear Prediction Cepstral Coefficients

In addition to the MFCC feature, the input speech signal is also parameterized in terms of LPCC, which we believe is able to provide complementary information to the MFCC features. Similar to that of the MFCC feature, the LPCC feature is extracted from the pre-emphasized speech signal using a 30 ms Hamming window with 10 ms overlap between frames. For each of the speech frame, an 18th order linear prediction analysis is performed using the autocorrelation method. Finally, 18 cepstral coefficients are derived from the LP coefficients. Dynamic information of the features is added by appending delta features, resulting in a 36-dimensional LPCC feature vector. Note that we do not include delta-delta features. Preliminary experiment on the NIST 2001 SRE dataset shows that a better performance can be achieved with the current setting.

2.3 Temporal Discrete Cosine Transform

In MFCC features, the delta and delta-delta features capture short-term dynamic information in the interval ranging from 50 to 100 ms. However, this interval is insufficient for longer term “high-level” features like prosodic gestures, and syllable usage. TDCT encodes the long-term dynamic of the cepstral features by taking the DCT over several frames [6]. Fig. 2 illustrates the TDCT features computation procedure. Each cepstral coefficient is considered as an independent signal which is windowed in blocks of length *B*. DCT is applied on each block, and the lowest *L* DCT coefficients, which contain most of the energy, are retained. Suppose we have *M* coefficients in the MFCC feature vector, the DCT coefficients can be stacked to form a long vector of dimensionality $M \times L$. The next TDCT vector is computed by advancing the block by one frame. Experimental results show that a block size of *B* = 8 frames, and *L* = 3 for the DCT, give the best performance on the NIST 2001 SRE dataset [6]. The resulting TDCT feature vector has a dimension of 36×3 = 108, and corresponds to a total time span of 250ms.

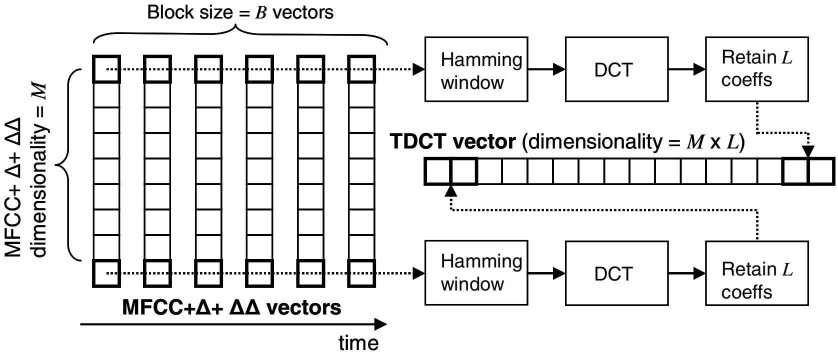


Fig. 2. Illustration of the TDCT features computation [6]

2.4 Voice Activity Detection

An energy-based voice activity detector (VAD) is applied after feature extraction. The VAD decides which feature vectors correspond to speech portions of the signal and which correspond to non-speech portions (i.e., silence and background noise). In particular, we use a GMM with 64 components to model the energy distribution of the speech frames pertaining to each of the two classes. The GMMs are trained beforehand using the development set of the NIST 2001 SRE corpus. The decision is then made through a likelihood ratio test, whereby speech frames with their energy having a higher likelihood with the speech GMM are retained, while those having a higher likelihood with the non-speech GMM are discarded. Recall that a TDCT feature vector is derived from a block of *B* MFCC feature vectors. If most of the MFCC facture vectors in a certain block belong to speech portion, then the TDCT feature vector derived from that specific block can be determined to be corresponding to speech portion. In our implementation, a TDCT feature vector is retained if more than 40% of the MFCC feature vectors in the block belong to speech portion. Finally, mean subtraction and variance normalization are applied to the outputs of the VAD to produce zero mean, unit variance MFCC, LPCC, and TDCT features.

3 Speaker Modeling and Pattern Matching

Given a speech utterance represented in terms of spectral feature vectors, as described in the previous section, the next step is to model the speaker specific information embedded in the given set of feature vectors. Two different approaches to speaker modeling and verification, as listed below, are employed in our system.

3.1 GMM-UBM

The GMM-UBM subsystems in Fig. 1 uses the standard set-up described in [7, 8]. A GMM is a weighted combination of a finite number of Gaussian distributions in the following form

$$p(\mathbf{x} \mid \lambda) = \sum_{k=1}^{K} w_k \, p_k(\mathbf{x}), \tag{1}$$

where w_k is the mixture weight associated with the *k*th Gaussian component given by

$$p_k(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} \left|\boldsymbol{\Sigma}_k\right|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}-\boldsymbol{\mu}_k)\right\}. \tag{2}$$

In the above equations, each of the Gaussian densities is parameterized by a $D \times 1$ mean vector $\boldsymbol{\mu}_k$ and a $D \times D$ covariance matrix $\boldsymbol{\Sigma}_k$, where D is the dimension of the feature vector $\mathbf{x}$. The mixture weights of all the K mixture components are by definition ≥ 0 and have to satisfy the constraint $\sum_{k=1}^{K} w_k = 1$. Collectively, the parameters of the mixture density, i.e., $\lambda = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$ for $k = 1, 2, \ldots, K$, represent a speaker model in the feature space of $\mathbf{x}$.

For a given test segment $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, the average log likelihood of the speaker model λ for the test segment, assuming that the feature vectors $\mathbf{x}_n$ are independent, is given by

$$\log p(X \mid \lambda) = \frac{1}{N} \sum_{n=1}^{N} \log p(\mathbf{x}_n \mid \lambda). \tag{3}$$

Notice that log-likelihood value is divided by *N,* which essentially normalizes out the duration effects of test segments with different length. The final score is then taken as a log likelihood ratio, as follows

$$\log \ \mathrm{LR} = \log p(X \mid \lambda) - \log p(X \mid \lambda_{\mathrm{UBM}}), \tag{4}$$

where λ_{UBM} is the *universal background model* (UBM) that represents a background set of speaker models. For computational simplicity, we use fast GMM-UBM scoring algorithm [8] using only the top 20 mixture components. It should be emphasized that the fast scoring algorithm makes sense only if the target model is adapted from the background model, as explained below.

In the training phase, speech segments from several background speakers are combined to train a UBM, thereby allowing the UBM to represent the speaker-independent distribution of features. The parameters of the UBM λ_{UBM} are estimated by maximum likelihood estimation, using the *expectation-maximization* (EM) algorithm. A speaker model λ is then derived by adapting the parameters of the UBM λ_{UBM} using the speech segment from the speaker by means of maximum *a posteriori* (MAP) training [8]. For numerical reasons, the covariance matrices pertaining to the Gaussian components are assumed to be diagonal.

3.2 Spectral SVM

SVM is a two-class classifier. For a given set of training samples with positive and negative labels, the SVM models the hyperplane that separates the two classes of samples. In the context of speaker verification, SVM models the boundary between a speaker and a set of background speakers that represent the population of impostors expected during recognition. The idea is different from the GMM-UBM, which models the distribution of the two classes. Furthermore, SVMs are non-probabilistic and use a different training philosophy compared to GMM. With a proper fusion strategy, both classifiers would complement each other in speaker recognition task [10].

The spectral SVM classifier in Fig. 1 closely follows the work reported in [9, 10], which greatly relies on polynomial expansion and the *generalized linear discriminant sequence* (GLDS) kernel. The central element of the GLDS kernel is a kernel inner product matrix defined as follows

$$\mathbf{R} \equiv E\{\mathbf{b}(\mathbf{x})\mathbf{b}^T(\mathbf{x})\}, \tag{5}$$

where $\mathbf{b}(\mathbf{x})$ denotes the polynomial expansion of the feature vector $\mathbf{x}$. For example, the second-order polynomial expansion of a two-dimensional vector $\mathbf{x} \equiv [x_1, x_2]^T$ is given by $\mathbf{b}(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_1x_2, x_2^2]^T$. For computational simplicity, it is customary to assumed that the matrix $\mathbf{R}$ is diagonal, i.e., $\mathbf{R} \approx diag[\mathbf{r}] = \mathbf{\Lambda}$, where the vector $\mathbf{r}$ is given by

$$\mathbf{r} = \frac{1}{M}\sum_{m=1}^{M} diag[\mathbf{b}(\mathbf{x}_m)\mathbf{b}^T(\mathbf{x}_m)]. \tag{6}$$

In the above equation, $\{\mathbf{x}_m\}_{m=1}^{M}$ denotes a pool of M feature vectors from all the non-target background speakers, and $diag[\,.\,]$ denotes the operation forming a diagonal matrix from a column vector and vice versa.

During enrollment, all the utterances in the background and the utterance for the current speaker under training are represented in terms of average expanded feature vectors in the following form

$$\mathbf{b}_{av} = \left[\frac{1}{N} \sum_{n=1}^{N} \mathbf{b}(\mathbf{x}_n) \right], \tag{7}$$

where N denotes the length of any specific utterance. These average expanded feature vectors are then normalized in the form $\boldsymbol{\Lambda}^{-1/2}\mathbf{b}_{av}$, assigned with appropriate label (i.e., +1 for target speaker, -1 for other competing speakers in the background), and finally used for SVM training. The output of the training is a set of support vectors $\mathbf{b}_i$, weights α_i, and a bias d. A speaker model $\mathbf{w}$ is then obtained by collapsing all the support vectors, as follows

$$\mathbf{w} = \left(\boldsymbol{\Lambda}^{-1/2} \sum_{i=1}^{l} \alpha_i t_i \mathbf{b}_i \right) + \mathbf{d}, \tag{8}$$

where $\mathbf{d} = [d,\ 0,\ \ldots,\ 0]^T$ and l denotes the number of support vectors resulted from the discriminative training. In the verification phase, for a given test segment $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, and a hypothesized speaker $\mathbf{w}$, the classifier score is obtained as the inner product between the speaker model $\mathbf{w}$ and the average expanded feature vector $\mathbf{b}_{av}$ pertaining to the test segment X, as follows

$$\text{score} = \mathbf{w}^T \mathbf{b}_{av}. \tag{9}$$

4 System Specifications

Given the approaches described in Section 2 and Section 3, four separate subsystems are constructed forming an ensemble of classifiers, as illustrated in Fig. 1. The four classifiers are (i) MFCC GMM-UBM, (ii) TDCT GMM-UBM, (iii) MFCC SVM, and (iv) LPCC SVM. For a given speech utterance, pattern matching is performed in the individual classifier, and a final score is obtained by combining the scores from all the subsystems. The specifications of the subsystems and fusion strategy are described below. The specifications presented below are obtained through numerous experiments carried using the development set of the CSLP SRE corpus, and some other corpora like CCC-VPR2C2005-6000 (*CCC 2-channel corpus for voiceprint recognition 2005 – 6000 speakers*) and NIST SRE corpus [4].

4.1 GMM-UBM

We have two separate GMM-UBM subsystems. The first one is based on MFCC, whereas the second one uses the new TDCT features described in Section 2.3. The UBMs are trained from the development set of the CSLP SRE corpus, which is guaranteed to be disjoint with the evaluation set [3].

Separate UBMs are used for the single-channel and cross-channel tasks. For single-channel task, we derive a 768-component UBM by training independently two channel-dependent UBMs of size 512 and 256 components, respectively, for landline and cellular channel types. The final UBM model is obtained by aggregating the Gaussian components of the two UBMs, and normalizing the mixture weights so that they sum to one. It should be noted that, channel-dependent UBM is not applicable here because channel-type information is not available for the evaluation data. On the other hand, a different composition is used for cross-channel task. In particular, the UBM has 768 components (1024 components for TDCT GMM-UBM) with 512 components trained from the landline data, and the remaining 256 components (512 components for TDCT GMM-UBM) trained from cellular data. The speaker models are then obtained by adapting the UBM parameters towards the speaker's training data using MAP adaptation principle. Therefore, the speaker models have the same number of Gaussian components with the UBM.

4.2 Spectral SVM

Two different sets of acoustic spectral features, namely MFCC and LPCC, are used thereby forming two separate SVM subsystems. The background or anti-speaker data consist of 4000 utterances extracted from CCC-VPR2C2005-6000. The evaluation set (for text-independent verification and identification tasks) of the CSLP SRE is derived from the CCC-VPR2C2005-1000, which is a subset of the CCC-VPR2C2005-6000 corpus. The CCC-VPR2C2005-1000 subset is discarded from the CCC-VPR2C2005-6000 beforehand so that the 4000 utterances used as the background would not overlap with the evaluation data.

Similar background data is used for the single-channel and cross-channel tasks. For each utterance in the background and for the target speaker, an average expanded feature vector is created. All monomials up to order 3 are used, resulting in a feature space expansion from 36 to 9139 in dimension. These average expanded feature vectors are used in the SVM training. The commonly available SVMTorch [11] is used for this purpose. The result of the training is a vector $\mathbf{w}$ of dimension 9139 which represents the desired target speaker model.

Test normalization (T-norm) method [12] is used to normalize the score. A collection of 500 cohort models are derived from development set of the CSLP SRE corpus. Scores from the cohort models are used to normalize a hypothesized speaker score for a given test segment. Score normalization is accomplished by subtracting the mean and dividing by the standard deviation of the scores produced by the cohort models in response to a given test segment. In order to obtain an accurate estimation of the mean and standard deviation parameters, the population of the cohort models has to be large enough. Furthermore, cohort models have to closely resemble the target speaker models. We believe that it is the best to establish the cohort models from the development set of the CSLP 2006 SRE.

4.3 Subsystems Integration

For a given speech utterance and a hypothesized speaker, pattern matching is performed separately in the four classifiers, giving rise to a 4-dimensional score vector. A final score is then derived from the score vector through a multilayer perceptron (MLP) neural network. The scores from all the subsystems are normalized to zero mean and unit variance before passing to the neural network. The MLP has 100 hidden neurons and one output neuron with sigmoid activation function. Conjugate gradient algorithm is used for the neural network training.

The development set of the CSLP SRE corpus is used to train two neural networks for score fusion, one for the single-channel verification and identification tasks, and the other one for cross-channel verification and identification tasks.

For speaker verification, the threshold (for the true/false decision) is set at a point whereby the following detection cost function (DCF) is minimized:

$$C_{\text{DET}} = C_{\text{Miss}} \times P_{\text{Miss}} \times P_{\text{Target}} + C_{\text{FalseAlarm}} \times P_{\text{FalseAlarm}} \times \left(1 - P_{\text{Target}}\right), \quad (10)$$

where P_{Miss} and $P_{\text{FalseAlarm}}$ are miss and false-alarm probabilities, respectively, and the parameters $C_{\text{Miss}} = 10$, $C_{\text{FalseAlarm}} = 1$, and $P_{\text{Target}} = 0.05$ are as indicated in the evaluation plan [3].

The speaker identification task is handled through a ranking and pruning procedure. First, a MLP score is derived for each pair of test sample and model. For each test sample, we rank the corresponding trial models with their MLP scores in descending order. Second, we extract all the pairs of test sample and its top-best matching model, rank them in descending order. The top 50% of the pairs are selected as the genuine test trials.

5 Evaluation Results

Fig. 3 and Fig. 4 depict the *detection error tradeoff* (DET) curves of the individual subsystems for the single-channel and cross-channel verification tasks, respectively. As mentioned earlier, these subsystems are fused at the score level using a neural network classifier. The neural networks are trained using the provided development set. The results of fusion are shown in Fig. 3 and Fig. 4 as well. The characteristics of the development set matches well with that of the evaluation set thereby giving a satisfactory fusion result when the trained neural networks are used for the evaluation dataset. The final decision thresholds for the verification tasks are also determined using the development set. On the other hand, the thresholds for the identification tasks is set according to 1:1 in-set and out-of-set ratio stated in the evaluation plan [3]. That is, the speaker identification tasks are performed in an open set manner.

Table 2 summarizes the performance of our submission to the CSLP 2006 SRE based on the actual DCF value and the *identification correctness rate* [3] for verification and identification tasks, respectively. As expected, channel mismatch makes the recognition tasks more difficult. The degradation in performance can be observed from both the DCF value and the identification correctness rate.

Table 3 summarizes the equal-error rates (EERs) and the minimum DCF values for the individual and fused scores. Clearly, the subsystems fuse in a complementary way reducing error rates substantially. Taking the LPCC SVM as baseline, the fused systems give relative EER improvements of 52% and 22% for single-channel and cross-channel conditions, respectively. On the other hand, the relative improvements in minimum DCF for single-channel and cross-channel verification tasks are 57% and

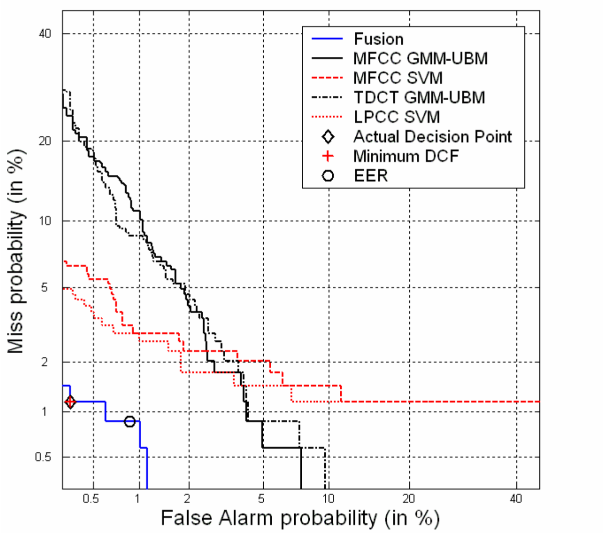


Fig. 3. DET curves for single-channel verification task

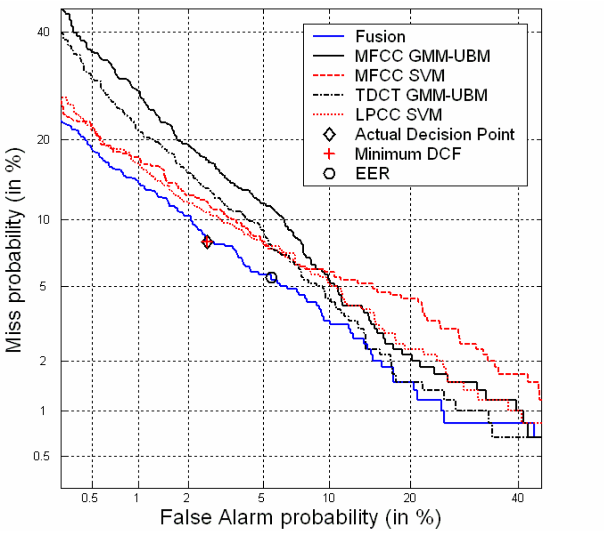


Fig. 4. DET curves for cross-channel verification task

18%, respectively. The gains in performance are due both to the different features (MFCC, LPCC, and TDCT) and the different speaker modeling techniques (SVM and GMM). From the DET curves, it can be noted that SVM and GMM complement each other at different threshold values. In particular, SVM performs best at high threshold values (i.e., upper left corner), while GMM dominates at low threshold values (i.e.,

Table 2. Performance of IIR submission to the 2006 CSLP SRE based on the DCF value and the identification correctness rate.

	Actual DCF value (×100)	Identification Correctness Rate
Single-Channel Verification Task	0.90	
Cross-Channel Verification Task	6.42	
Single-Channel Identification Task		97.16%
Cross-Channel Identification Task		86.45%

Table 3. Comparison of EER and minimum DCF for IIR individual subsystems/final system in speaker verification tasks

System	Single-channel verification task		Cross-channel verification task	
	EER (%)	Min DCF (×100)	EER (%)	Min DCF (×100)
MFCC GMM-UBM	2.54	3.44	7.70	10.22
MFCC SVM	2.31	2.31	6.71	8.10
TDCT GMM-UBM	2.85	3.89	6.69	8.68
LPCC SVM	1.81	2.09	7.03	7.79
Fusion	0.86	0.90	5.50	6.42

lower left corner). It can also be observed that SVM performs best with LPCC features. On the other hand, GMM performs best with MFCC and TDCT features for single and cross-channel tasks, respectively, mainly due to the difference in the UBMs. Further research into optimizing features for each of the modeling techniques should be carried out.

6 Conclusions

A description of a speaker recognition system has been presented as it was developed for the CSLP 2006 SRE. Our submission was built upon three different acoustic spectral features and two different speaker modeling techniques giving rise to four subsystems, namely, MFCC GMM-UBM, TDCT GMM-UBM, MFCC SVM, and LPCC SVM. These subsystems were combined at the score level through a MLP neural network in a complementary way. The fused system achieved an EER of 0.86% and 5.50% for single-channel and cross-channel verification tasks, respectively. Promising results were also obtained for identification tasks, where identification rates of 97.16% and 86.45% were obtained under single-channel and cross-channel conditions, respectively. The SRE results confirm a successful design and implementation of speaker recognition system. Nevertheless, continuous effort that makes use of the common platform provided by the CSLP SRE event should be carried out.

References

1. S. Furui, “Speaker verification,” in *Digital Signal Processing Handbook*, V. K. Madisetti and D. B. Williams, Eds. Boca Raton: CRC Press LLC, 1999.
2. T. F. Quatieri, *Discrete-time speech signal processing: principles and practice*. Upper-Sadder River, NJ: Prentice-Hall, 2002.
3. *Evaluation Plan for ISCSLP’2006 Special Session on Speaker Recognition*, Chinese Corpus Consortium, Apr. 2006.
4. D. A. van Leeuwen, A. F. Martin, M. A. Przybocki, and J. S. Bouten, “NIST and NFI-TNO evaluations of automatic speaker recognition,” *Computer Speech and Language*, vol. 20, pp. 128-158, 2006.
5. S. B. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, no. 4, Aug. 1980.
6. T. H. Kinnunen, C. W. E. Koh, L. Wang, H. Li and E. S. Chng, “Shifted delta cepstrum amd temporal discrete cosine transform features in speaker verification,” accepted for presentation in *International Symposium on Chinese Spoken Language Processing,* 2006.
7. F. Bimbot, J. F. Bonastre, C. Fredouille, G. Gravier, I. Margin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska, and D. A. Reynolds, “A tutorial on text-indepent speaker verification,” *Eurasip Journal on Applied Signal Processing*, vol. 4, pp. 430-451, 2004.
8. D.A. Reynolds, T.F. Quatieri and R.B. Dunn, “Speaker verification using adapted Gaussian mixture models”, *Digital Signal Processing*, vol. 10, pp. 19-41, 2000.
9. W. M. Campbell, “Generalized linear discriminant sequence kernels for speaker recognition,” in *Proc. ICASSP*, pp. 161-164, 2002.
10. W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, “Support vector machines for speaker and language recognition,” *Computer Speech and Language*, vol. 20, no. 2-3, pp. 210-229, 2006.
11. R. Collobert and S. Bengio, “SVMTorch: support vector machines for large-scale regression problems,” Journal of Machine Learning Research, vol. 1, pp. 143-160, 2001.
12. Auckenthaler, M. Carey, and H. Lloyd-Thomas, “Score normalization for text-independent speaker verification systems,” *Digital Signal Processing*, vol. 10, no 1-3, pp. 42-54, Jan 2000.

A Novel Alternative Hypothesis Characterization Using Kernel Classifiers for LLR-Based Speaker Verification

Yi-Hsiang Chao[1,2], Hsin-Min Wang[1], and Ruei-Chuan Chang[1,2]

[1] Institute of Information Science, Academia Sinica, Taipei
[2] Department of Computer Science, National Chiao Tung University, Hsinchu
{yschao, whm}@iis.sinica.edu.tw, rc@cc.nctu.edu.tw

Abstract. In a log-likelihood ratio (LLR)-based speaker verification system, the alternative hypothesis is usually ill-defined and hard to characterize a priori, since it should cover the space of all possible impostors. In this paper, we propose a new LLR measure in an attempt to characterize the alternative hypothesis in a more effective and robust way than conventional methods. This LLR measure can be further formulated as a non-linear discriminant classifier and solved by kernel-based techniques, such as the Kernel Fisher Discriminant (KFD) and Support Vector Machine (SVM). The results of experiments on two speaker verification tasks show that the proposed methods outperform classical LLR-based approaches.

Keywords: Speaker verification, Log-likelihood ratio, Kernel Fisher Discriminant, Support Vector Machine.

1 Introduction

In essence, the speaker verification task is a hypothesis testing problem. Given an input utterance U, the goal is to determine whether U was spoken by the hypothesized speaker or not. The log-likelihood ratio (LLR)-based [1] detector is one of the state-of-the-art approaches for speaker verification. Consider the following hypotheses:

H_0: U is from the hypothesized speaker,
H_1: U is not from the hypothesized speaker.

The LLR test is expressed as

$$L(U) = \log \frac{p(U \mid H_0)}{p(U \mid H_1)} \begin{cases} \geq \theta & \text{accept } H_0 \\ < \theta & \text{accept } H_1 \text{ (i.e., reject } H_0\text{)}, \end{cases} \tag{1}$$

where $p(U \mid H_i)$, $i = 0, 1,$ is the likelihood of the hypothesis H_i given the utterance U, and θ is the threshold. H_0 and H_1 are, respectively, called the null hypothesis and the alternative hypothesis. Mathematically, H_0 and H_1 can be represented by parametric models denoted as λ and $\bar{\lambda}$, respectively; $\bar{\lambda}$ is often called an anti-model. Though H_0 can be modeled straightforwardly using speech utterances from the hypothesized speaker, H_1 does not involve any specific speaker, and thus lacks explicit data for

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 506 – 517, 2006.

modeling. Many approaches have been proposed to characterize H_1, and various LLR measures have been developed. We can formulate these measures in the following general form [2]:

$$L(U) = \log \frac{p(U \mid \lambda)}{\Psi(p(U \mid \lambda_1), p(U \mid \lambda_2), \ldots, p(U \mid \lambda_N))}, \tag{2}$$

where $\Psi(\cdot)$ is some function of the likelihood values from a set of so-called background models $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$. For example, the background model set can be obtained from N representative speakers, called a cohort [8], which simulates potential impostors. If $\Psi(\cdot)$ is an average function [1], the LLR can be written as

$$L_1(U) = \log p(U \mid \lambda) - \log \left\{ \frac{1}{N} \sum_{i=1}^{N} p(U \mid \lambda_i) \right\}. \tag{3}$$

Alternatively, the average function can be replaced by various functions, such as the maximum [3], i.e.,

$$L_2(U) = \log p(U \mid \lambda) - \max_{1 \le i \le N} \log p(U \mid \lambda_i), \tag{4}$$

or the geometric mean [4], i.e.,

$$L_3(U) = \log p(U \mid \lambda) - \frac{1}{N} \sum_{i=1}^{N} \log p(U \mid \lambda_i). \tag{5}$$

A special case arises when $\Psi(\cdot)$ is an identity function and $N = 1$. In this instance, a single background model is usually trained by pooling all the available data, which is generally irrelevant to the clients, from a large number of speakers. This is called the world model or the Universal Background Model (UBM) [2]. The LLR in this case becomes

$$L_4(U) = \log p(U \mid \lambda) - \log p(U \mid \Omega), \tag{6}$$

where Ω denotes the world model.

However, none of the LLR measures developed so far has proved to be absolutely superior to any other, since the selection of $\Psi(\cdot)$ is usually application and training data dependent. In particular, the use of a simple function, such as the average, maximum, or geometric mean, is a heuristic that does not involve any optimization process. The issues of selection, size, and combination of background models motivate us to design a more comprehensive function, $\Psi(\cdot)$, to improve the characterization of the alternative hypothesis. In this paper, we first propose a new LLR measure in an attempt to characterize H_1 by integrating all the background models in a more effective and robust way than conventional methods. Then, we formulate this new LLR measure as a non-linear discriminant classifier and apply kernel-based techniques, including the Kernel Fisher Discriminant (KFD) [6] and Support Vector Machine (SVM) [7], to optimally separate the LLR samples of the null hypothesis from those of the alternative hypothesis. Speaker verification experiments conducted on both the XM2VTSDB database and the

ISCSLP2006 speaker recognition evaluation database show that the proposed methods outperform classical LLR-based approaches.

The remainder of this paper is organized as follows. Section 2 describes the analysis of the alternative hypothesis in our approach. Sections 3 and 4 introduce the kernel classifiers used in this work and the formation of the characteristic vector by background model selection, respectively. Section 5 contains our experiment results. Finally, in Section 6, we present our conclusions.

2 Analysis of the Alternative Hypothesis

First of all, we redesign the function $\Psi(\cdot)$ in Eq. (2) as

$$\Psi(\mathbf{u}) = (p(U \mid \lambda_1)^{\alpha_1} \cdot p(U \mid \lambda_2)^{\alpha_2} \cdot \ldots \cdot p(U \mid \lambda_N)^{\alpha_N})^{1/(\alpha_1+\alpha_2+\ldots+\alpha_N)}, \tag{7}$$

where $\mathbf{u} = [p(U \mid \lambda_1), p(U \mid \lambda_2), \ldots, p(U \mid \lambda_N)]^T$ is an Nx1 vector and α_i is the weight of the likelihood $p(U \mid \lambda_i)$, i = 1,2,…, N. This function gives N background models different weights according to their individual contribution to the alternative hypothesis. It is clear that Eq. (7) is equivalent to a geometric mean function when $\alpha_i = 1$, i = 1,2,…, N. If some background model λ_i contrasts with an input utterance U, the likelihood $p(U \mid \lambda_i)$ may be extremely small, and thus cause the geometric mean to approximate zero. In contrast, by assigning a favorable weight to each background model, the function $\Psi(\cdot)$ defined in Eq. (7) may be less affected by any specific background model with an extremely small likelihood. Therefore, the resulting score for the alternative hypothesis obtained by Eq. (7) will be more robust and reliable than that obtained by a geometric mean function. It is also clear that Eq. (7) will reduce to a maximum function when $\alpha_{i^*} = 1$, $i^* = \arg\max_{1 \le i \le N} \log p(U \mid \lambda_i)$; and $\alpha_i = 0$, $\forall i \neq i^*$.

By substituting Eq. (7) into Eq. (2) and letting $w_i = \alpha_i / (\alpha_1 + \alpha_2 + \ldots + \alpha_N)$, i = 1,2,…, N, we obtain

$$\begin{aligned} L(U) &= \log \frac{p(U \mid \lambda)}{p(U \mid \lambda_1)^{w_1} \cdot p(U \mid \lambda_2)^{w_2} \cdot \ldots \cdot p(U \mid \lambda_N)^{w_N}} \\ &= \log \left(\left(\frac{p(U \mid \lambda)}{p(U \mid \lambda_1)} \right)^{w_1} \cdot \left(\frac{p(U \mid \lambda)}{p(U \mid \lambda_2)} \right)^{w_2} \cdot \ldots \cdot \left(\frac{p(U \mid \lambda)}{p(U \mid \lambda_N)} \right)^{w_N} \right) \\ &= w_1 \log \frac{p(U \mid \lambda)}{p(U \mid \lambda_1)} + w_2 \log \frac{p(U \mid \lambda)}{p(U \mid \lambda_2)} + \ldots + w_N \log \frac{p(U \mid \lambda)}{p(U \mid \lambda_N)} \\ &= \mathbf{w}^T \mathbf{x} \begin{cases} \ge \theta & \text{accept} \\ < \theta & \text{reject,} \end{cases} \end{aligned} \tag{8}$$

where $\mathbf{w} = [w_1, w_2 \ldots, w_N]^T$ is an Nx1 weight vector and $\mathbf{x}$ is an $N \times 1$ vector in the space R^N, expressed by

$$\mathbf{x} = [\log \frac{p(U \mid \lambda)}{p(U \mid \lambda_1)}, \log \frac{p(U \mid \lambda)}{p(U \mid \lambda_2)}, \ldots, \log \frac{p(U \mid \lambda)}{p(U \mid \lambda_N)}]^T . \tag{9}$$

The implicit idea in Eq. (9) is that the speech utterance U can be represented by a characteristic vector **x**.

If we replace the threshold θ in Eq. (8) with a bias b, the equation can be rewritten as

$$L(U) = \mathbf{w}^T \mathbf{x} + b = f(\mathbf{x}) , \tag{10}$$

where $f(\mathbf{x})$ forms a so-called linear discriminant classifier. This classifier translates the goal of solving an LLR measure into the optimization of **w** and b, such that the utterances of clients and impostors can be separated. To realize this classifier, three distinct data sets are needed: one for generating each client's model, one for generating the background models, and one for optimizing **w** and b. Since the bias b plays the same role as the decision threshold θ of the conventional LLR-based detector defined in Eq. (1), which can be determined through a trade-off between false acceptance and false rejection, the main goal here is to find **w**.

3 Kernel Classifiers

Intuitively, $f(\mathbf{x})$ in Eq. (10) can be solved via linear discriminant training algorithms [9]. However, such methods are based on the assumption that the observed data of different classes is linearly separable, which is obviously not feasible in most practical cases with nonlinearly separable data. To solve this problem more effectively, we propose using a kernel-based nonlinear discriminant classifier. It is hoped that data from different classes, which is not linearly separable in the original input space R^N, can be separated linearly in a certain higher dimensional (maybe infinite) feature space F via a nonlinear mapping Φ. Let $\Phi(\mathbf{x})$ denote a vector obtained by mapping **x** from R^N to F. Then, the objective function, based on Eq. (10), can be re-defined as

$$f(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) + b , \tag{11}$$

which constitutes a linear discriminant classifier in F.

In practice, it is difficult to determine the kind of mapping that would be applicable; therefore, the computation of $\Phi(\mathbf{x})$ might be infeasible. To overcome this difficulty, a promising approach is to characterize the relationship between the data samples in F, instead of computing $\Phi(\mathbf{x})$ directly. This is achieved by introducing a kernel function $k(\mathbf{x}, \mathbf{y})=<\Phi(\mathbf{x}),\Phi(\mathbf{y})>$, which is the dot product of two vectors $\Phi(\mathbf{x})$ and $\Phi(\mathbf{y})$ in F. The kernel function $k(\cdot)$ must be symmetric, positive definite and conform to Mercer's condition [7]. A number of kernel functions exist, such as the simplest dot product kernel function $k(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y}$, and the very popular Radial Basis Function (RBF) kernel $k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2 / 2\sigma^2)$ in which σ is a tunable parameter. Existing techniques, such as KFD [6] or SVM [7], can be applied to implement Eq. (11).

3.1 Kernel Fisher Discriminant (KFD)

Suppose the *i*-th class has n_i data samples, $\mathbf{X}_i = \{\mathbf{x}_1^i, .., \mathbf{x}_{n_i}^i\}$, i = 1, 2. The goal of the KFD is to find a direction **w** in the feature space *F* such that the following Fisher's criterion function *J*(**w**) is maximized:

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b^\Phi \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w^\Phi \mathbf{w}}, \tag{12}$$

where $\mathbf{S}_b^\Phi$ and $\mathbf{S}_w^\Phi$ are, respectively, the between-class scatter matrix and the within-class scatter matrix defined as

$$\mathbf{S}_b^\Phi = (\mathbf{m}_1^\Phi - \mathbf{m}_2^\Phi)(\mathbf{m}_1^\Phi - \mathbf{m}_2^\Phi)^T \tag{13}$$

and

$$\mathbf{S}_w^\Phi = \sum_{i=1,2} \sum_{\mathbf{x} \in \mathbf{X}_i} (\Phi(\mathbf{x}) - \mathbf{m}_i^\Phi)(\Phi(\mathbf{x}) - \mathbf{m}_i^\Phi)^T, \tag{14}$$

where $\mathbf{m}_i^\Phi = (1/n_i)\sum_{s=1}^{n_i} \Phi(\mathbf{x}_s^i)$, and i = 1, 2, is the mean vector of the *i*-th class in *F*. Let $\mathbf{X}_1 \cup \mathbf{X}_2 = \{\mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_l\}$ and $l = n_1 + n_2$. Since the solution of **w** must lie in the span of all training data samples mapped in *F* [6], **w** can be expressed as

$$\mathbf{w} = \sum_{j=1}^{l} \alpha_j \Phi(\mathbf{x}_j). \tag{15}$$

Let $\boldsymbol{\alpha}^T = [\alpha_1, \alpha_2, \ldots, \alpha_l]$. Accordingly, Eq. (11) can be re-written as

$$f(\mathbf{x}) = \sum_{j=1}^{l} \alpha_j k(\mathbf{x}_j, \mathbf{x}) + b. \tag{16}$$

Our goal therefore changes from finding **w** to finding **α**, which maximizes

$$J(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^T \mathbf{M} \boldsymbol{\alpha}}{\boldsymbol{\alpha}^T \mathbf{N} \boldsymbol{\alpha}}, \tag{17}$$

where **M** and **N** are computed by

$$\mathbf{M} = (\boldsymbol{\eta}_1 - \boldsymbol{\eta}_2)(\boldsymbol{\eta}_1 - \boldsymbol{\eta}_2)^T \tag{18}$$

and

$$\mathbf{N} = \sum_{i=1,2} \mathbf{K}_i (\mathbf{I}_{n_i} - \mathbf{1}_{n_i}) \mathbf{K}_i^T, \tag{19}$$

respectively, where $\boldsymbol{\eta}_i$ is an lx1 vector with $(\boldsymbol{\eta}_i)_j = (1/n_i)\sum_{s=1}^{n_i} k(\mathbf{x}_j, \mathbf{x}_s^i)$, $\mathbf{K}_i$ is an lxn_i matrix with $(\mathbf{K}_i)_{js} = k(\mathbf{x}_j, \mathbf{x}_s^i)$, $\mathbf{I}_{n_i}$ is an n_ixn_i identity matrix, and $\mathbf{1}_{n_i}$ is an n_ixn_i matrix with all entries equal to $1/n_i$. Following [6], the solution for **α**, which maximizes *J*(**α**) defined in Eq. (17), is the leading eigenvector of $\mathbf{N}^{-1}\mathbf{M}$.

3.2 Support Vector Machine (SVM)

Alternatively, Eq. (11) can be solved with an SVM, the goal of which is to seek a separating hyperplane in the feature space F that maximizes the margin between classes. Following [7], **w** is expressed as

$$\mathbf{w} = \sum_{j=1}^{l} y_j \alpha_j \Phi(\mathbf{x}_j), \tag{20}$$

which yields

$$f(\mathbf{x}) = \sum_{j=1}^{l} y_j \alpha_j k(\mathbf{x}_j, \mathbf{x}) + b, \tag{21}$$

where each training sample $\mathbf{x}_j$ belongs to one of the two classes identified by the label $y_j \in \{-1,1\}$, j=1, 2,..., l. We can find the coefficients α_j by maximizing the objective function,

$$Q(\boldsymbol{\alpha}) = \sum_{j=1}^{l} \alpha_j - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j), \tag{22}$$

subject to the constraints,

$$\sum_{j=1}^{l} y_j \alpha_j = 0, \text{ and } 0 \le \alpha_j \le C, \ \forall j, \tag{23}$$

where C is a penalty parameter [7]. The problem can be solved using quadratic programming techniques [10]. Note that most α_j are equal to zero, and the training samples associated with non-zero α_j are called *support vectors*. A few support vectors act as the key to deciding the optimal margin between classes in the SVM. An SVM with a dot product kernel function is known as a Linear SVM.

4 Formation of the Characteristic Vector

In our experiments, we use B+1 background models, consisting of B cohort set models and one world model, to form the characteristic vector **x** in Eq. (9); and B cohort set models for $L_1(U)$ in Eq. (3), $L_2(U)$ in Eq. (4), and $L_3(U)$ in Eq. (5). Two cohort selection methods [1] are used in the experiments. One selects the B closest speakers to each client; and the other selects the B/2 closest speakers to, plus the B/2 farthest speakers from, each client. The selection is based on the speaker distance measure [1], computed by

$$d(\lambda_i, \lambda_j) = \log \frac{p(X_i \mid \lambda_i)}{p(X_i \mid \lambda_j)} + \log \frac{p(X_j \mid \lambda_j)}{p(X_j \mid \lambda_i)}, \tag{24}$$

where λ_i and λ_j are speaker models trained using the i-th speaker's utterances X_i and the j-th speaker's utterances X_j, respectively. Two cohort selection methods yield the following two (B+1)x1 characteristic vectors:

$$\mathbf{x} = [\tilde{p}_0(U)\ \tilde{p}_1^c(U)\ \ldots\ \tilde{p}_B^c(U)]^T \tag{25}$$

and

$$\mathbf{x} = [\tilde{p}_0(U)\ \tilde{p}_1^c(U)\ \ldots\ \tilde{p}_{B/2}^c(U)\ \tilde{p}_1^f(U)\ \ldots\ \tilde{p}_{B/2}^f(U)]^T, \tag{26}$$

where $\tilde{p}_0(U) = \log p(U \mid \lambda) / p(U \mid \Omega)$, $\tilde{p}_i^c(U) = \log p(U \mid \lambda) / p(U \mid \lambda_{\text{closest } i})$, and $\tilde{p}_i^f(U) = \log p(U \mid \lambda) / p(U \mid \lambda_{\text{farthest } i})$. $\lambda_{\text{closest } i}$ and $\lambda_{\text{farthest } i}$ are the i-th closest model and the i-th farthest model of the client model λ , respectively.

5 Experiments

We evaluate the proposed approaches on two databases: the XM2VTSDB database [11] and the ISCSLP2006 speaker recognition evaluation (ISCSLP2006-SRE) database [12].

For the performance evaluation, we adopt the Detection Error Tradeoff (DET) curve [13]. In addition, the NIST Detection Cost Function (DCF) [14], which reflects the performance at a single operating point on the DET curve, is also used. The DCF is defined as

$$C_{DET} = C_{Miss} \times P_{Miss} \times P_{Target} + C_{FalseAlarm} \times P_{FalseAlarm} \times (1 - P_{Target}), \tag{27}$$

where P_{Miss} and $P_{FalseAlarm}$ are the miss probability and the false-alarm probability, respectively, C_{Miss} and $C_{FalseAlarm}$ are the respective relative costs of detection errors, and P_{Target} is the *a priori* probability of the specific target speaker. A special case of the DCF is known as the Half Total Error Rate (HTER), where C_{Miss} and $C_{FalseAlarm}$ are both equal to 1, and P_{Target} = 0.5, i.e., $\text{HTER} = (P_{Miss} + P_{FalseAlarm})/2$.

5.1 Evaluation on the XM2VTSDB Database

The first set of speaker verification experiments was conducted on speech data extracted from the XM2VTSDB multi-modal database [11]. In accordance with "Configuration II" described in [11], the database was divided into three subsets: "Training", "Evaluation", and "Test". In our experiments, we used the "Training" subset to build the individual client's model and the world model, and the "Evaluation" subset to estimate the decision threshold θ in Eq. (1) and the parameters **w** and b in Eq. (11). The performance of speaker verification was then evaluated on the "Test" subset. As shown in Table 1, a total of 293 speakers[1] in the database were divided into 199 clients, 25 "evaluation impostors", and 69 "test impostors". Each speaker participated in four, recording sessions at approximately one-month intervals, and each recording session consisted of two shots. In a shot, every speaker was prompted to utter three sentences

[1] We discarded 2 speakers (ID numbers 313 and 342) because of partial data corruption.

Table 1. Configuration of the XM2VTSDB speech database

Session	Shot	199 clients	25 impostors	69 impostors
1	1	Training	Evaluation	Test
	2			
2	1			
	2			
3	1	Evaluation		
	2			
4	1	Test		
	2			

"0 1 2 3 4 5 6 7 8 9", "5 0 6 9 2 8 1 3 7 4", and "Joe took father's green shoe bench out". Using a 32-ms Hamming-windowed frame with 10-ms shifts, each utterance (sampled at 32 kHz) was converted into a stream of 24-order feature vectors, each consisting of 12 Mel-scale cepstral coefficients [5] and their first time derivatives.

We used 12 (2×2×3) utterances/speaker from sessions 1 and 2 to train the individual client's model, represented by a Gaussian Mixture Model (GMM) [1] with 64 mixture components. For each client, the other 198 clients' utterances from sessions 1 and 2 were used to generate the world model, represented by a GMM with 256 mixture components; 20 speakers were chosen from these 198 clients as the cohort. Then, we used 6 utterances/client from session 3, and 24 (4×2×3) utterances/evaluation-impostor, which yielded 1,194 (6×199) client samples and 119,400 (24×25×199) impostor samples, to estimate θ, $\mathbf{w}$, and b. However, because a kernel-based classifier can be intractable when a large number of training samples is involved, we reduced the number of impostor samples from 119,400 to 2,250 using a uniform random selection method. In the performance evaluation, we tested 6 utterances/client in session 4 and 24 utterances/test-impostor, which produced 1,194 (6×199) client trials and 329,544 (24×69×199) impostor trials.

5.1.1 Experiment Results

We implemented the proposed LLR system in four ways: KFD with Eq. (25) ("KFD_w_20c"), KFD with Eq. (26) ("KFD_w_10c_10f"), SVM with Eq. (25) ("SVM_w_20c"), and SVM with Eq. (26) ("SVM_w_10c_10f"). Both SVM and KFD used an RBF kernel function with $\sigma = 5$. For the performance comparison, we used five systems as our baselines: 1) $L_1(U)$ with the 20 closest cohort models ("*L*1_20c"), 2) $L_1(U)$ with the 10 closest cohort models plus the 10 farthest cohort models ("*L*1_10c_10f"), 3) $L_2(U)$ with the 20 closest cohort models ("*L*2_20c"), 4) $L_3(U)$ with the 20 closest cohort models ("*L*3_20c"), and 5) $L_4(U)$ ("*L*4").

Fig. 1 shows the results of the baseline systems tested on the "Evaluation" subset in DET curves [13]. We observe that the curves "*L*1_10c_10f" and "*L*4" are better than the others. Thus, in the second experiment, we focused on the performance improvements of our proposed LLR systems over these two baselines.

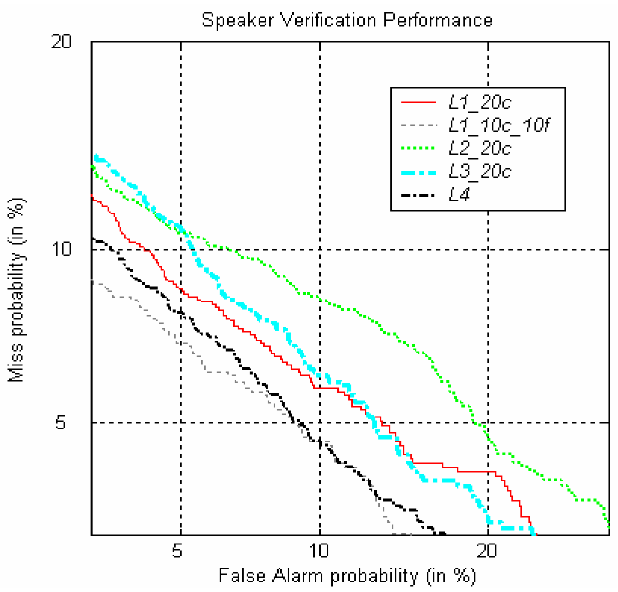


Fig. 1. Baselines: DET curves for the XM2VTSDB "Evaluation" subset

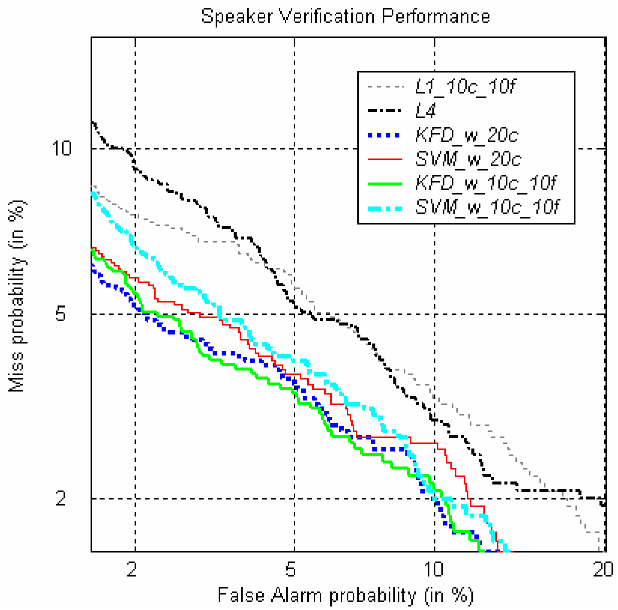


Fig. 2. Best baselines vs. our proposed LLR systems: DET curves for the XM2VTSDB "Test" subset

Table 2. HTERs for "Evaluation" and "Test" subsets (The XM2VTSDB task)

	min HTER for "Evaluation"	HTER for "Test"
*L*1_20c	0.0676	0.0535
*L*1_10c_10f	0.0589	0.0515
*L*2_20c	0.0776	0.0635
*L*3_20c	0.0734	0.0583
*L*4	0.0633	0.0519
KFD_w_20c	0.0247	0.0357
SVM_w_20c	0.0320	0.0414
KFD_w_10c_10f	0.0232	0.0389
SVM_w_10c_10f	0.0310	0.0417

Fig. 2 shows the results of our proposed LLR systems versus the baseline systems evaluated on the "Test" subset. It is clear that the proposed LLR systems, including KFD and SVM, outperform the baseline LLR systems, while KFD performs better than SVM.

An analysis of the results based on the HTER is given in Table 2. For each approach, the decision threshold, θ or b, was used to minimize the HTER on the "Evaluation" subset, and then applied to the "Test" subset. From Table 2, we observe that, for the "Test" subset, a 30.68% relative improvement was achieved by "KFD_w_20c", compared to "*L*1_10c_10f" – the best baseline system.

5.2 Evaluation on the ISCSLP2006-SRE Database

We participated in the text-independent speaker verification task of the ISCSLP2006 Speaker Recognition Evaluation (SRE) plan [12]. The database, which was provided by Chinese Corpus Consortium (CCC) [15], contained 800 clients. The length of the training data for each client ranged from 21 seconds to 1 minute and 25 seconds; the average length was approximately 37.06 seconds.

We sorted the clients according to the length of their training data in descending order. For the first 100 clients, we cut two 4-second segments from the end; and for the remaining 700 clients, we cut one 4-second segment from the end, as the "Evaluation" data to estimate θ, **w**, and b. For each client, the remaining training data was used for "Training" to build that client's model. In the implementation, all the "Training" data was pooled to train a UBM [2] with 1,024 mixture components. Then, the mean vectors of each client's GMM were adapted from the UBM by his/her "Training" data. In the evaluation stage, each client was treated as an "evaluation impostor" of the other 799 clients. In this way, we had 900 (2×100+700) client samples and 719,100 (900×799) impostor samples. We applied all the client samples and 2,400 randomly selected impostor samples to estimate **w** of the kernel classifiers. According to the evaluation plan, the ratio of true clients to imposters in the "Test" subset should be approximately 1:20. Therefore, we applied the 900 client samples and 18,000 randomly selected impostor samples to estimate the decision threshold, θ or b. The "Test" data consisted of 5,933 utterances.

The signal processing front-end was same as that applied in the XM2VTSDB task.

5.2.1 Experiment Results

Fig. 3 shows the results of the proposed LLR system using KFD with Eq. (26) and B = 100 ("KFD_w_50c_50f") versus the baseline GMM-UBM [2] system tested on 5,933 "Test" utterances in DET curves. The proposed LLR system clearly outperforms the baseline GMM-UBM system. According to the ISCSLP2006 SRE plan, the performance is measured by the NIST DCF with $C_{Miss} = 10$, $C_{FalseAlarm} = 1$, and P_{Target} =0.05. In each system, the decision threshold, θ or b, was selected to minimize the DCF on the "Evaluation" data, and then applied to the "Test" data. The minimum DCFs for the "Evaluation" data and the associated DCFs for the "Test" data are given in Table 3. We observe that "KFD_w_50c_50f" achieved a 34.08% relative improvement over "GMM-UBM".

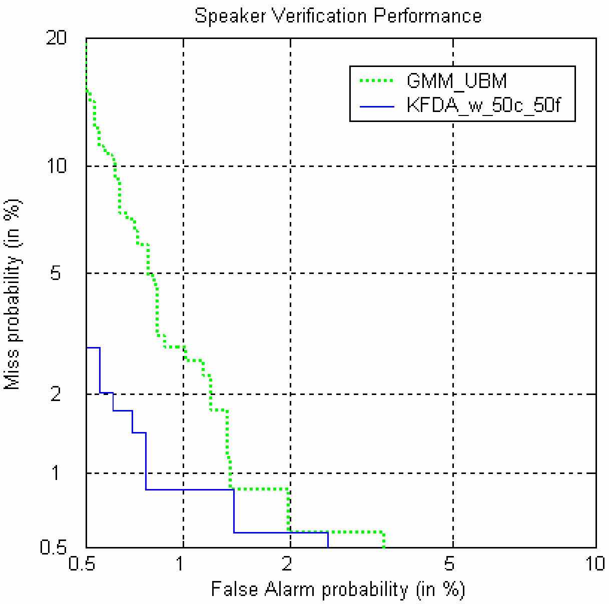


Fig. 3. DET curves for the ISCSLP2006-SRE "Test" subset

Table 3. DCFs for "Evaluation" and "Test" subsets (The ISCSLP2006-SRE task)

	min DCF for "Evaluation"	DCF for "Test"
GMM-UBM	0.0129	0.0179
KFD_w_50c_50f	0.0067	0.0118

6 Conclusions

We have presented a new LLR measure for speaker verification that improves the characterization of the alternative hypothesis by integrating multiple background models in a more effective and robust way than conventional methods. This new LLR measure is formulated as a non-linear classification problem and solved by using kernel-based classifiers, namely, the Kernel Fisher Discriminant and Support Vector Machine, to optimally separate the LLR samples of the null hypothesis from those of the alternative hypothesis. Experiments, in which the proposed methods were applied to two speaker verification tasks, showed notable improvements in performance over classical LLR-based approaches. Finally, it is worth noting that the proposed methods can be applied to other types of data and hypothesis testing problems.

Acknowledgments. This work was funded by the National Science Council, Taiwan, under Grant: NSC94-2213-E-001-009.

References

1. Reynolds, D. A.: Speaker Identification and Verification using Gaussian Mixture Speaker Models. Speech Communication, Vol.17. (1995) 91-108
2. Reynolds, D. A., Quatieri, T. F., Dunn, R. B.: Speaker Verification using Adapted Gaussian Mixture Models. Digital Signal Processing, Vol. 10. (2000) 19-41
3. Higgins, A., Bahler, L., Porter, J.: Speaker Verification using Randomized Phrase Prompting. Digital Signal Processing, Vol. 1. (1991) 89-106
4. Liu, C. S., Wang, H. C., Lee, C. H.: Speaker Verification using Normalized Log-Likelihood Score. IEEE Trans. Speech and Audio Processing, Vol. 4. (1996) 56-60
5. Huang, X., Acero, A., Hon, H. W.: Spoken Language Processing. Prentics Hall, New Jersey (2001)
6. Mika, S., Rätsch, G., Weston, J. Schölkopf, B., Müller, K. R.: Fisher Discriminant Analysis with Kernels. Neural Networks for Signal Processing IX. (1999) 41-48
7. Burges, C.: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery, Vol.2. (1998) 121-167
8. Rosenberg, A. E., Delong, J., Lee, C. H., Juang, B. H., Soong, F. K.: The use of Cohort Normalized Scores for Speaker Verification. Proc. ICSLP (1992)
9. Duda, R. O., Hart, P. E., Stork, D. G.: Pattern Classification. 2nd edn. John Wiley & Sons, New York (2001)
10. Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York (1998)
11. Luettin, J., Maître, G.: Evaluation Protocol for the Extended M2VTS Database (XM2VTSDB). IDIAP-COM 98-05, IDIAP (1998)
12. Chinese Corpus Consortium (CCC): Evaluation Plan for ISCSLP’2006 Special Session on Speaker Recognition (2006)
13. Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M.: The DET Curve in Assessment of Detection Task Performance. Proc. Eurospeech (1997)
14. http://www.nist.gov/speech/tests/spk/index.htm
15. http://www.CCCForum.org

Speaker Verification Using Complementary Information from Vocal Source and Vocal Tract

Nengheng Zheng, Ning Wang, Tan Lee, and P.C. Ching

Department of Electronic Engineering, The Chinese University of Hong Kong,
Hong Kong
{nhzheng, nwang, tanlee, pcching}@ee.cuhk.edu.hk

Abstract. This paper describes a speaker verification system which uses two complementary acoustic features: Mel-frequency cepstral coefficients (MFCC) and wavelet octave coefficients of residues (WOCOR). While MFCC characterizes mainly the spectral envelope, or the formant structure of the vocal tract system, WOCOR aims at representing the spectro-temporal characteristics of the vocal source excitation. Speaker verification experiments carried out on the ISCSLP 2006 SRE database demonstrate the complementary contributions of MFCC and WOCOR to speaker verification. Particularly, WOCOR performs even better than MFCC in single channel speaker verification task. Combining MFCC and WOCOR achieves higher performance than using MFCC only in both single and cross channel speaker verification tasks.

1 Introduction

Automatic speaker recognition is a process of automatically determining a person's identity based on the intrinsic characteristics of his/her voice. It is one of the most important speech processing techniques and can be implemented in many real-world applications, e.g. access control, telephone banking, public security, and so on. The state-of-the-art speaker recognition systems typically employ the Mel-frequency cepstral coefficients (MFCC) as the representative acoustic features and the statistical modeling techniques (e.g. the Gaussian Mixture Models, GMM) for pattern matching [1], [2]. The MFCC+GMM architecture has achieved very good performance (even better than recognition by human [3]) in laboratory experiments or in some small scale, sophistically controlled applications. However, in most of the real-world applications, the speaker recognition performances are far from being reliable due to environmental distortions, shortage of speech materials for statistical modeling and testing, increased speaker characteristics overlapping in large scale systems, and so on.

Many efforts have been devoted to developing new techniques to improve the reliability and robustness of speaker recognition system in real-world applications. One technique is to extract new speaker-specific features to supplement the conventional MFCC features to improve the recognition accuracy. For example, the use of fundamental frequency (F0) and prosodic features for speaker recognition has been reported in many papers [4], [5]. However, the pitch value

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 518–528, 2006.

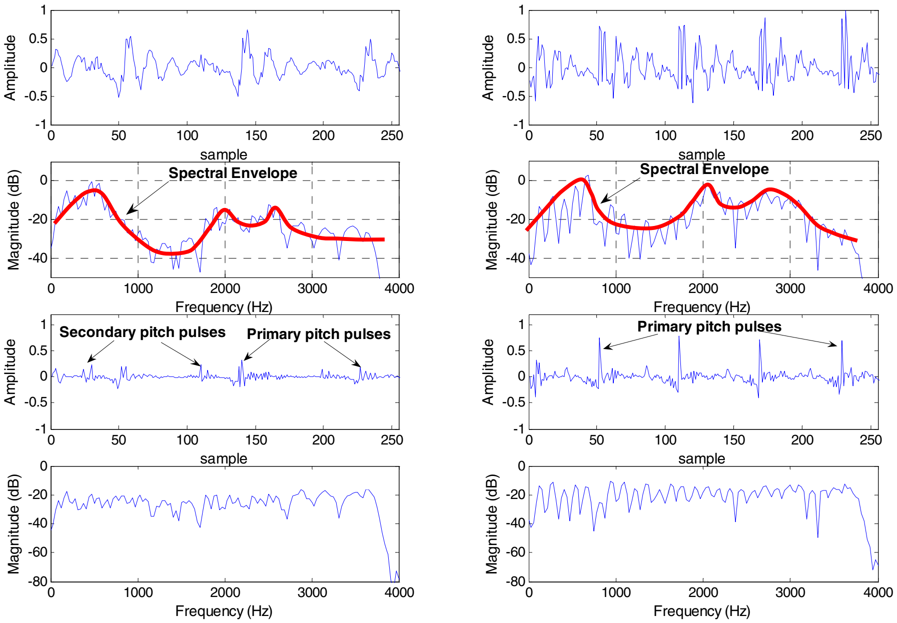


Fig. 1. Examples of speech waveforms and LP residual signals of two male speakers. Left: Speaker A; Right: Speaker B; Top to bottom: speech waveforms, Fourier spectra of speech signals, LP residual signals and Fourier spectra of LP residual signals.

has large intra-speaker variations, including the fluctuation along a single utterance and the long-term intra-speaker variations between the training and testing utterances recorded at different time. The large intra-speaker variation restricts the contribution of pitch related features for speaker recognition. In recent years, generating high-level features, e.g. the speaking style, to supplement the low-level acoustic features has become a hot topic in speaker recognition [6], [7]. However, learning the speaker-specific speaking style generally requires a large amount of training speech materials, and the effective feature extraction technique has not been thoroughly exploited.

This paper describes a speaker recognition system which integrates two complementary acoustic features for speaker verification. Besides the conventional MFCC parameters, the system also incorporates the so called Wavelet Octave Coefficient of Residues (WOCOR) as a complementary feature to MFCC. As known, human speech can be modeled as the convolutional output of a vocal source excitation signal and the impulse response of the vocal tract filter system [8]. The MFCC parameters are derived mainly to represent the spectral envelope, or the formant structure of the vocal tract system [9]. The WOCOR parameters, on the other hand, are derived from the linear predictive (LP) residual signal and aim at characterizing the spectro-temporal characteristics of the vocal source excitation [10]. Figure 1 shows the speech waveforms of the vowel /a/ uttered by two male speakers, their corresponding LP residual signals and

Fourier spectra of speech and residual signals. As shown, there are noticeable differences between the residual signals of the two speakers. In addition to the significant difference between their pitch periods, the residual signal of speaker B shows much stronger periodicity than that of speaker A. For speaker A, the magnitudes of the secondary pulses are relatively higher. For both speakers, the short-time Fourier transforms of their residual signals give nearly flat spectra. Although the harmonic structures of the spectra reflect the periodicity, the pitch pulses related time-frequency properties cannot be easily extracted from the Fourier spectra. To characterize the time-frequency characteristics of the pitch pulses, wavelet transform is applied to generate the WOCOR parameters, which characterize the spectro-temporal characteristics of the residual signal.

MFCC and WOCOR contain complementary speaker-specific information since they characterize two physiologically different information sources in speech production. In this paper, the effectiveness of combining MFCC and WOCOR for speaker verification is evaluated over the database provided for the ISCSLP2006 Speaker Recognition Evaluation (SRE) by the Chinese Corpus Consortium (CCC). Evaluation results show that the proposed system achieves higher recognition performance than the MFCC based system in both single and cross channel conditions.

2 Database Description

The experiments carried out in this paper are part of the ISCSLP 2006 SRE tasks. Two tasks, i.e. text-independent single-channel speaker verification and text-independent cross-channel speaker verification, are presented.

The database for the evaluation is divided into development and evaluation sets. The development data are used for system development, e.g. determining the decision thresholds and training the fusion parameters. It contains 300 male speakers. Each speaker has 2 utterances corresponding to land-line (PSTN) and cellular-phone (GSM only) channels respectively. Each utterance is divided into several segments with at least one segment longer than 30 seconds for speaker model training and several shorter segments for testing. There are totally 2151 segments in the development data set.

All the final results presented are carried out with the evaluation data set. For the single-channel verification task, the evaluation data set contains 800 male speakers, each having a training segment longer than 30 ms. There are 3157 testing segments with various durations. There are total 5939 verification tests with ratio of true-speaker and impostor tests being about 1:20. For a specific speaker, the training and testing segments are from the same channel. For the cross-channel verification task, there are also 800 speakers, each having a training segment longer than 30 ms. There are 3827 testing segments and 11788 verification tests with ratio of true-speaker and impostor tests being about 1:20. For each speaker, the training and testing utterances must come from different channels. There is no overlaps between the speakers of development and evaluation data sets.

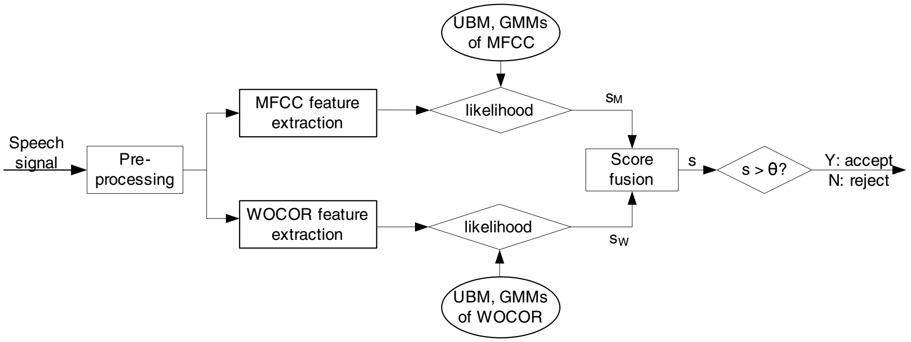


Fig. 2. Block diagram of the system using MFCC and WOCOR for speaker verification

3 Speaker Verification System with MFCC and WOCOR

Figure 2 gives the block diagram of the system using MFCC and WOCOR for speaker verification. In the pre-processing stage, the speech signal is first pre-emphasized with a first order filter $H(z) = 1 - 0.97z^{-1}$. Then an energy-based voice activity detection (VAD) technique is applied to remove the silence portion. The speech signal is passed through for MFCC and WOCOR generation respectively. For each feature set, speaker models are trained with the UBM-GMM technique [11] in the training stage. In the testing stage, for each testing utterance, two matching scores are calculated from MFCC and WOCOR respectively and they are combined to give a final score for verification decision.

3.1 Generating the Vocal Tract Related Features MFCC

The extraction of MFCC follows the standard procedures as described in [9]:

1) Short-time Fourier transform is applied every 10 ms with a 30 ms Hamming window.

2) The magnitude spectrum is warped with a Mel-scale filter bank that consists of 26 filters, which emulates the frequency resolution of human auditory system. Log-magnitude of each filter output is calculated.

3) Discrete cosine transform (DCT) is applied to the filter bank output.

The MFCC feature vector has 39 components, including the first 12 cepstral coefficients, the log energy, as well as their first and second order time derivatives. Since the speech data used in our experiments were recorded via public telephone networks, the method of cepstral mean normalization (CMN) is applied to eliminate the convolutional channel distortion [12].

3.2 Generating the Vocal Source Related Features WOCOR

While the MFCC features are extracted from both voiced and unvoiced speech signal, the WOCOR features are extracted from only the voiced speech. This is because WOCOR are derived mainly to capture the time-frequency characteristics of the pitch pulses of the LP residual signal. The excitation signal for unvoiced speech in the source-filter model is approximated as a random noise [8]. We believe that such a noise-like signal carries little speaker-specific information in the time-frequency domain. The process of extracting the proposed WOCOR features is formulated in the following steps.

1) *Voicing decision and pitch extraction.* Pitch extraction and voicing status decision are done by the Robust Algorithm for Pitch Tracking (RAPT) algorithm [13]. Only voiced speech is kept for subsequent processing.

2) *LP inverse filtering.* LP analysis of order 12 is performed every 30 ms with Hamming window. The filter coefficients a_k are computed using the autocorrelation method [8]. The residual signal $e(n)$ is obtained by inverse filtering the speech signal $s(n)$, i.e.

$$e(n) = s(n) - \sum_{k=1}^{12} a_k s(n-k). \tag{1}$$

To reduce the intra-speaker variation, the amplitude of $e(n)$ is normalized to be within the interval [-1, 1].

3) *Pitch-synchronous windowing.* With the pitch periods estimated in step 1, pitch pulses in the residual signal are located. For each pitch pulse, pitch-synchronous wavelet analysis is applied with a Hamming window of two pitch periods long. Let t_{i-1}, t_i and t_{i+1} denote the locations of three successive pitch pulses. The analysis window for the pitch pulse at t_i spans from t_{i-1} to t_{i+1}. The windowed residual signal is denoted as $e_h(n)$.

4) *Wavelet transform of the residual signal.* The wavelet transform of $e_h(n)$ is computed as

$$w(a,b) = \frac{1}{\sqrt{|a|}} \sum_n e_h(n) \Psi^*(\frac{n-b}{a}) \tag{2}$$

where $a = \{2^k | k = 1, 2 \cdots, 6\}$ and $b = 1, 2 \cdots, N$, and N is the window length. $\Psi(n)$ is the 4th-order Daubechies wavelet basis function. a and b are called the scaling parameter and the translation parameter respectively [14]. Assuming a signal bandwidth of 4000 Hz, the signal is decomposed into six sub-bands at different octave levels: 2000 - 4000 Hz ($k = 1$), 1000 - 2000 Hz, $\cdots$, 62.5 - 125 Hz ($k = 6$). At a specific sub-band, the time-varying characteristics within the analysis window are measured as the translation parameter b changes.

5) *Generating the feature parameters.* We now have six octave groups of wavelet coefficients, i.e.,

$$W_k = \left\{ w(2^k, b) \, | b = 1, 2, \cdots, N \right\}, k = 1, 2, ..., 6 \tag{3}$$

Each octave group of coefficients is divided evenly into M sub-groups, i.e.,

$$\begin{aligned} &W_k^M(m) = \left\{ w(2^k, b) \, | b \in (m-1:m] \times (N/M) \right\} \\ &m = 1, 2, \cdots, M \end{aligned} \tag{4}$$

The 2-norm of each sub-group of coefficients is computed to be a feature parameter. As a result, the complete feature vector is composed as,

$$\mathrm{WOCOR}_M = \left\{ \left\| W_k^M(m) \right\| \middle| \begin{matrix} m = 1, 2, \cdots, M \\ k = 1, 2, \cdots, 6 \end{matrix} \right\} \tag{5}$$

where $\| \cdot \|$ denotes the 2-norm operation.

In the case of $M = 1$, all the coefficients of a sub-band are combined to form a single feature parameter, and therefore, all temporal information are lost. On the other hand, if $M = N$, each coefficient is included as an individual component in the feature vector. This may introduce too much unnecessary detail so that the features become less discriminative. From statistical modeling point of view, a relatively low feature dimension is also desirable. Our previous experiments showed that generally, $M = 4$ gives good enough performance and larger M does not improve the performance significantly [10].

To summarize, given a speech utterance, a sequence of WOCOR feature vectors are obtained by pitch-synchronous analysis of the LP residual signal. Each feature vector consists of 24 components, which capture useful spectro-temporal characteristics of the residual signal.

3.3 Integrating MFCC and WOCOR for Speaker Verification

In general, the integration of different information sources for speaker verification can be done at (i) feature level, (ii) score level, and (iii) decision level. In this system, the integration is done at score level. Score level is preferred because the matching scores are easily available and contain sufficient information for distinguishing different speakers. Also, as described above, WOCOR and MFCC are calculated in different time-resolution, it is difficult to concatenate them together. Therefore, the feature level fusion is not applicable. To do the score level fusion, we first train two models, one for MFCC and the other for WOCOR, for each speaker. We adopt the state-of-the-art GMM-UBM approach for statistical speaker modeling and speaker verification [11]. That is, for each feature set, a universal background model (UBM), which is a large scale Gaussian mixture model (GMM) with 1024 mixtures, is first built using the development data described in Sec. 2. Then for each target speaker, a speaker model is adapted from the UBM using the respective training data. In the verification stage, given a testing utterance, two log-likelihood ratio (LLR) scores are obtained from the MFCC and WOCOR streams respectively. For each stream, the LLR score is given as

$$s_i = \log P(s_i|\lambda_{\mathrm{c},i}) - \log P(s_i|\lambda_{\mathrm{U},i}) \tag{6}$$

where $P(s_i|\lambda_{\mathrm{c},i})$ and $P(s_i|\lambda_{\mathrm{U},i})$ denote the likelihoods given by the GMMs of the claimed speaker and the UBM respectively. The subscript i denotes the different features: $i = 1$ for MFCC and $i = 2$ for WOCOR. For verification decision, a final score is obtained by a linear combination of s_1 and s_2, i.e.,

$$s = w_t s_1 + (1 - w_t) s_2 \tag{7}$$

where the fusion w_t is trained using the development data.

3.4 Evaluation Metrics

If s is higher than a preset threshold θ, the claimant is accepted. Otherwise it is rejected. There are two types of errors: false acceptance (FA) of an impostor and false rejection (FR) of the genuine speaker. With different values of θ, different FA and FR rates can be attained. This leads to a detection error trade-off (DET) curve for evaluating the system performance [15]. Each point on the DET curve corresponds to a specific value of θ, with the horizontal and vertical coordinates being the FA rate and the FR rate respectively. A DET curve closer to the original point corresponds to a system with higher verification performance.

Another commonly adopted evaluation metric is the detection cost function (DCF) which is defined as a weighted sum of FR and FA probabilities [16]:

$$DCF = C_{FR} \cdot P_{FR} \cdot P_{Target} + C_{FA} \cdot P_{FA} \cdot (1 - P_{Target}) \tag{8}$$

where C_{FR} and C_{FA} are the detection error costs of false rejecting a target speaker and false accepting an impostor. P_{FR} and P_{FA} are the probabilities of false rejection and false acceptance. P_{Target} is the *prior* probability of testing utterances coming from the target speaker among all the testing utterances.

While the DET curve gives a general evaluation of the system showing how the FR can be traded off against FA in different decision thresholds, the DCF evaluates the system performance at specified decision conditions.

4 Experimental Results

4.1 Single Channel Speaker Verification

For the single channel speaker verification, the DET curves for the MFCC based system, the WOCOR based system and the combined system are given as in Fig. 3. As illustrated, the WOCOR based system performs even better than the MFCC based system. The combined system achieves much higher verification performance than that using only one feature. On each of the DET curves, the circle marked gives the FR rate and FA rate with the decision threshold θ, which is trained using the development data, of the corresponding system. The DCFs of the three systems with the respective θ are 4.74, 4.12 and 1.75, respectively. In comparison to the system using MFCC only, the combined system reduces the DCF by about 63% relatively.

4.2 Cross Channel Speaker Verification

Figure 4 illustrates the performances of the MFCC based system, WOCOR based system and the combined system in cross channel speaker verification. As illustrated, WOCOR performs much worse than MFCC, which means that WOCOR is very sensitive to channel mismatch between the training and testing segments. Even though, combining MFCC and WOCOR still improves the overall performance to a certain extent, as illustrated in the figure. The DCFs for the three systems are 14.07, 25.68 and 12.43 in this case. In comparison to the system using MFCC only, a relative 12% reduction of DCF has been achieved by the proposed system.

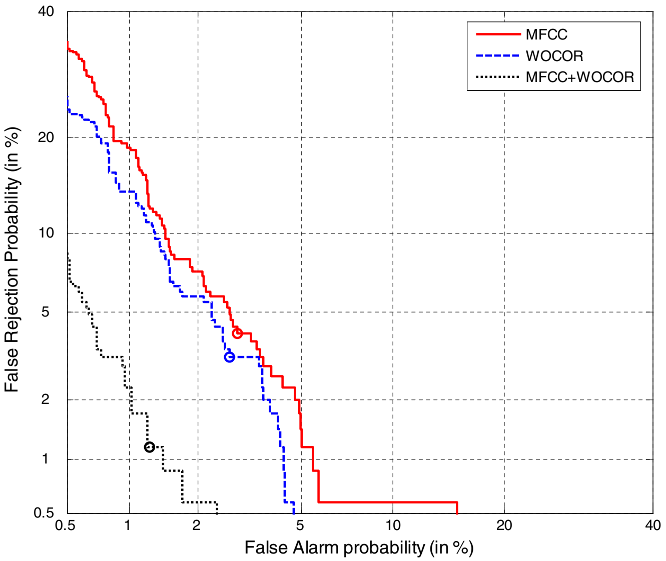


Fig. 3. DET curves for the MFCC based, WOCOR based and combined systems in single channel speaker verification

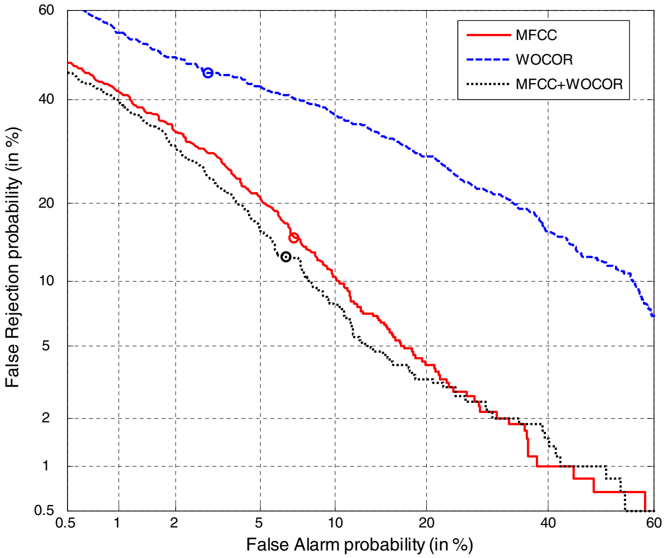


Fig. 4. DET curves for the MFCC based, WOCOR based and combined systems in cross channel speaker verification

5 Discussion

With the primary goal of identifying different speech sounds, MFCC features characterize mainly the spectral envelope of a quasi-stationary speech segment and provide important cues for phonetic classification. The spectral envelope corresponds to the vocal tract filter, which determines the articulation of sounds. Therefore, MFCC depends largely on linguistic content being spoken. On the other hand, WOCOR captures the vocal source information, which is related mainly to the periodicity and voice quality. Therefore, WOCOR is relatively less dependent on the content. This property of WOCOR is very useful in text-independent speaker recognition. As demonstrated in the single channel verification task, WOCOR performs better than MFCC. However, this task does not address the long-term intra-speaker variation because the training and testing segments of each speaker come from the same utterance. When the training and testing speeches are recorded in different time sessions, MFCC generally achieves higher performance than WOCOR, which implies that the vocal source excitation has larger degree of long-term intra-speaker variation than vocal tract system. Even though, combining MFCC (or LPCC) and WOCOR can still improve the recognition performance [10]. Our recent work showed that WOCOR is particularly useful in speaker segmentation task, which divides a speech utterance into homogenous segments containing speech of exactly one speaker [17].

Besides the long-term intra-speaker variation, another major challenge in speaker recognition is the mismatch between training and testing conditions, including the channel mismatch as addressed in the cross channel verification task. In this task, the training and testing segments come from two different channels, i.e. land-line (PSTN) and cellular phone (GSM). The frequency response of transmitting channel mainly changes the spectral envelope of speech signal and degrades the performance of MFCC in speaker recognition. Applying CMN on MFCC reduces the impact of channel mismatch to a large extent. On the other hand, land-line and cellular telephone networks use different speech coding techniques. Particularly, GSM cellular network adopts the code-excited linear prediction (CELP) coding technique. The parametric coding scheme of CELP changes the excitation signal of the speech [18]. As a result, the derived WOCOR parameters can not represent the exact vocal source characteristics of the original speaker. Nevertheless, in CELP, the excitation signal is selected from the codebook such that the re-synthesized speech is perceptually similar to the original one. Therefore, although the WOCOR parameters perform much worse than MFCC (with CMN) in cross channel task, they still contain some useful speaker information and can improve the recognition performance to a certain extent, as demonstrated in Fig. 4.

Although the sensitivity of WOCOR to channel mismatch restricts its applications in cross channel speaker verification. In some real-world applications, if training data from different channels are available, integrating MFCC and WOCOR can significantly improve the system performance without any knowledge of the channel conditions. We design a verification experiment with the ISCSLP2006 SRE development data to simulate this application scenario. In this

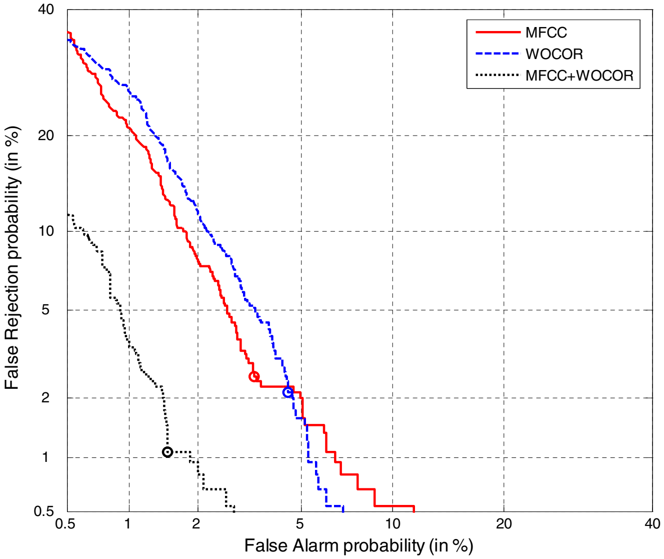


Fig. 5. DET curves for SV systems with 2 channel training data

experiment, a GMM is trained for each speaker using 2 speech segments, with one the land-line telephone speech and the other the cellular speech. The testing segments come from both channels without any prior knowledge of the channel types or any channel identification techniques implemented. The speaker verification performances of MFCC, WOCOR, and the combined systems are shown in Fig. 5. It is clear that using both MFCC and WOCOR significantly improves the verification performance.

6 Conclusion

This paper describes a speaker verification system using the conventional vocal tract related features MFCC and a newly proposed vocal source related features WOCOR. The two features have complementary contributions to speaker recognition. Experimental results show that WOCOR performs better than MFCC in single channel speaker verification task. Combining MFCC and WOCOR achieves much higher performance than using MFCC only in single channel speaker verification. Although WOCOR is more sensitive to channel mismatch, the combined system still improves the overall performance to a certain extent in the cross-channel speaker verification task.

Acknowledgements

This work is partly supported by a research grant awarded by the Research Grant Council of the Hong Kong SAR Government.

References

1. Campbell, J.P.: Speaker recognition: a tutorial. Proc. IEEE **85**(9) (1997) 1437–1462
2. Reynolds, D.A.: Speaker identification and verification using gaussian mixture speaker models. Speech Communication **17**(1) (1995) 91–108
3. Schmidt-Nielsen, A., Crystal, T.H.: Speaker verification by human listeners: Experiments comparing human and machine performance using the nist 1998 speaker evaluation data. Digital Signal Processing **10**(1-2) (2000) 249–266
4. Atal, B.S.: Automatic speaker recognition based on pitch contours. J. Acoust. Soc. Am. **52** (1972) 1687–1697
5. Sonmez, M.K., Heck, L., Weintraub, M., Shriberg, E.: A lognormal tied mixture model of pitch for prosody based speaker recognition. In: Proc. Eurospeech. (1997) 1391–1394
6. Reynolds, D., Andrews, W., Campbell, J., Navratil, J., Peskin, B., Adami, A., Jin, Q., Klusacek, D., Abramson, J., Mihaescu, R., Godfrey, J., Jones1, D., Xiang, B.: The SuperSID project: Exploiting high-level information for high-accuracy speaker recognition. In: Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing. (2003) 784–787
7. Campbell, W.M., Campbell, J.P., Reynolds, D.A., Jones, D.A., Leek, T.R.: High-level speaker verification with support vector machines. In: Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing. (2004) 73–76
8. Rabiner, L.R., Schafer, R.W.: Digital Processing of Speech Signals. Prentice Hall (1978)
9. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust., Speech, Signal Processing **28**(4) (1980) 357–366
10. Zheng, N.H., Ching, P.C., Lee, T.: Time frequency analysis of vocal source signal for speaker recognition. In: Proc. Int. Conf. on Spoken Language Processing. (2004) 2333–2336
11. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker verification using adapted gaussian mixture models. Digital Signal Processing **10**(1-3) (2000) 19–41
12. Atal, B.S.: Efectiveness of linear prediction characteristics of the speech wave for automatic speaker identication and verication. J. Acoust. Soc. Am. **55**(6) (1974) 1304–1312
13. Talkin, D.: A robust algorithm for pitch tracking (RAPT). Speech Coding and Synthesis (W. B. Kleijn and K. K. Paliwal Eds. Elsevier, 1995)
14. Daubechies, I.: Ten Lectures on Wavelets. Society for Industrial and Applied Mathematics, Philadelphia (1992)
15. Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M.: The DET curve in assessment of detection task performance. In: Proc. Eurospeech. (1997) 1895–1898
16. Martin, A., Przybocki, M.: The nist 1999 speaker recognition evaluation: An overview. Digital Signal Processing **10** (1-18) 2000
17. Chan, W., Lee, T., Zheng, N., Ouyang, H.: Use of vocal source features in speaker segmentation. In: Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing. (2006) 657–660
18. Campbell, J.P., Tremain, T.E., Welch, V.: The proposed federal standard 1016 4800 bps voice coder: CELP. Speech Technology (1990) 58–64

ISCSLP SR Evaluation, UVA–CS_es System Description. A System Based on ANNs*

Carlos E. Vivaracho

Dep. Informática, U. de Valladolid
cevp@infor.uva.es

Abstract. This paper shows a description of the system used in the ISCSLP06 Speaker Recognition Evaluation, text independent cross-channel speaker verification task. It is a discriminative Artificial Neural Network-based system, using the Non-Target Incremental Learning method to select world representatives. Two different training strategies have been followed: (i) to use world representative samples with the same channel type as the true model, (ii) to select the world representatives from a pool of samples without channel type identification. The best results have been achieved with the first alternative, but with the appearance of the additional problem of the true model channel type recognition. The system used in this task will also be shown.

1 Introduction

This work is concerned with the use of discriminative Artificial Neural Networks (ANNs) to deal with the speaker recognition problem.

The use of ANNs in Speaker Verification (SV), although very popular at the beginning of the 90s [13,14] with promising results [1], has decreased in recent years, where statistical classifiers, such as Gaussian Mixture Models (GMMs), or other discriminative classifiers, such as Support Vector Machines (SVM) [18], are more widely used.

In recent years, the SV task using ANNs has been mainly carried out by members of the *speech and vision laboratory* from the Indian Institute of Technology, Madras (India) [19][12], using Autoassociative Neural Networks (AANN) [8,3]. Other approaches can be found in [6], which uses a Locally Recurrent Probabilistic Neural Network, or in [9], using a Self Organizing Map (SOM).

Our perspective differs from these, since a Multilayer Perceptron (MLP) is used as a discriminative classifier, that is, as the estimator of the degree of membership of vector x_i to a class λ_C. The good performance of our MLP-based systems in the text independent cross-channel speaker verification task was shown in previous works [17][15]. Here, two different systems are presented:

- The so-called *System 1*, where the MLP is trained using world representative samples with the same channel type as the true model.

* This work has been supported by the Ministerio de Ciencia y Tecnología, Spain, under Project TIC2003-08382-C05-03.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 529–538, 2006.

– *System 2*, where the world representatives are selected from a pool of samples with all the channel types of the corpus.

System 2 is similar to that shown in [15], but with a different parameter extraction. The learning strategy followed in *System 1* has not been used in previous works. The advantage of this learning strategy is shown, as the final evaluation result is approximately 30% better than that achieved with System 2.

In order to recognize the channel type of the true model for selecting the world representatives with the same channel type, an MLP-based system was also used. The system is described in greater detail in Section 4.3.

An additional problem appears in the MLP training. The task approached is the so called two-class classification problem [7], since the classifier must classify a sample as belonging, or not, to a certain class called the Target Class (TC) (true model), and examples of the TC and the "Non-Target Class" (NTC) (impostors) are both available. If this task is approached by means of a discriminative classifier, it must be trained with examples from both the TC and the NTC.

The problem that appears is the imbalance in the datasets. A set of examples (or training set) is said to be imbalanced if one of the classes is represented by a small number of cases as compared to the other classes. In this situation, the inductive learning systems may have difficulties to learn the concept related with the minority class. This situation is found in the speaker verification task, where the number of the impostor class (NTC) training examples is much higher than those of the true speaker (TC).

Solutions to remedy the class imbalance problem have been proposed at both the data and algorithmic levels. Our solution concerns the former. At this level, the different solutions can be included in the following two forms of re-sampling [4][2]:

– **over-sampling** the minority class until the classes are approximately equally represented. This over-sampling can be accomplished with random replacement, directed replacement (in which no new samples are created but the choice of samples to replace is informed rather than random) and with informed generation of new examples.
– **under-sampling** the majority class until the classes are approximately equally represented. This method can be performed randomly or directed, where the choice of examples to eliminate is informed.

In the SV task the probability distribution of the examples is unknown and the generation of artificial samples is not recommended [11]. Thus, over-sampling by means of informed generation of new examples is not a good idea. With regard to the other over-sampling technique, replacement, it has shown good results in the past [5][17] with small databases and, thus, a small imbalance in the data sets. With the maturing of the problem, the databases became more realistic, increasing the level of class imbalance. Under these conditions minor class sample replacement significantly increases the training time, and in addition, an over-learning of the classifier was observed, decreasing the system performance. For these reasons our efforts have focused on the under-sampling solution.

The problem to be solved, under-sampling the major class, lies in selecting the most informative examples from a pool of representatives of the NTC, so as not to discard potentially useful information.

In [15] a new directed sampling algorithm called *Non-Target Incremental Learning* (NTIL) was proposed. This algorithm contains the classifier which directs the most discriminative NTC examples subset selection during the learning stage. Due to the effectiveness of this method in solving the problem, it is used in both *Systems 1* and *2*.

This paper is organized as follows. The heuristic solution NTIL to the imbalanced data set problem is shown in section 2. Section 3 gives a general description of the ANN-based SV system. The experiments performed with the development dataset are shown in section 4. The final system configuration and the results achieved in the text independent cross-channel speaker verification task can be seen in section 5. The conclusions are presented in section 6.

2 Non Target Incremental Learning Algorithm

To reduce notation, the NTC (impostor) samples used in classifier learning will be called *Training Subset* (TS), and the pool of examples of the NTC used for extracting the TS will be called the *Candidates for Training Subset* (CTS).

The proposed technique can be formulated as follows:

a) The size of the TS (*MaxTS*) and the ending criteria of the classifier training (*EndCri*) are selected.
b) The classifier is trained using the true training sample/s and one of those in the CTS until *EndCri* is reached. We begin with a preliminary, weak classifier.
c) The score of the classifier $S(X)$ is obtained for each of the remaining samples of the CTS, and the N with the highest values are chosen as the most similar (nearest) to the true model. If the classifier were used for classification, these N samples would be those with the highest probability of producing false acceptances. However, due to their discriminative capability, they are selected to train the classifier in the next phase.
d) The N samples chosen in step c) are included in the TS and the classifier is trained once again.
e) Steps c) and d) are repeated until *MaxIT* is reached.

The TS selection in each iteration of the previous technique can be seen, although not in the strict sense, as the N "not yet learned" samples.

To avoid random initialization of the TS, even the first TS sample (step b) is also heuristically selected as shown in [15].

3 General Description of the ANN-Based SV System

An MLP is trained per target speaker, using an Error Backpropagation learning algorithm. The general configuration of the system is as follows:

- Three layer MLP architecture, with 32 hidden units and one output unit.
- Non-linear sigmoid units are used in hidden and output layers.
- The desired outputs for input vectors belonging to the target speaker and TS are 1.0 and 0.0, respectively.
- The values of the learning coefficient and momentum are 0.01 and 0.95, respectively.
- The *EndCri* is to train the MLP over 75 epochs.

A more detailed description can be found in [17][16].

Although the assumptions to approximate the MLP output to a Bayesian a posteriori probability are not fulfilled [10], given an input vector x_i from a test (speech) sample $X = \{x_1, x_2, ..., x_M\}$, the output of the λ_C MLP, trained to verify the C target speaker identity, can be seen as the estimation of the membership degree of vector x_i to the class λ_C. This value is represented as $\Gamma(\lambda_C/x_i)$. Following this MLP output interpretation, in view of its meaning and values, even though it is not a real probability, it can be treated as such. Then, the "pseudo-probability" that a test sample X belongs to the speaker C will be:

$$\Gamma(\lambda_C/X) = \prod_{i=1}^{M} \Gamma(\lambda_C/x_i) \tag{1}$$

This value can be very low. So, to avoid loss of accuracy in its codification, the use of the logarithm is advisable:

$$\log(\Gamma(\lambda_C/X)) = \sum_{i=1}^{M} \log(\Gamma(\lambda_C/x_i)) \tag{2}$$

The result in eq. 2 is highly dependent on M. Additional use of the mean to avoid this dependence also allows the result to be bounded. Thus, the final score, $S(X)$, of the system per test sample X will be:

$$S(X) = \frac{1}{M} \sum_{i=1}^{M} \log(\Gamma(\lambda_{S_c}/x_i)) \tag{3}$$

In order to improve the system performance, the R262 rule [16] is used, modifying the previous calculation of the final score as follows:

$$S_R(X) = \frac{1}{N_R} \sum_{i=1}^{N_R} \log(\Gamma(\lambda_c/x_i)) \;\; \forall x_i/\Gamma(\lambda_c/x_i) \notin (0.2, 0.8) \tag{4}$$

Where N_R is the number of vectors x_i that verify the rule in (4).

As different classifiers, trained for different target speakers, give different score distributions, Z-Norm is accomplished to normalize these score distributions.

4 Experiments with the Development Data

Here, the main experiments performed with the development data and the conclusions extracted are shown.

4.1 Development Data Sample

The 300 speakers included in the development data were split into the following three different random subsets:

- Subset 1. Consisting of 100 speakers that will be used as true speakers. One sample (file) is randomly selected to train the MLP for each speaker, and the remaining ones (on average, approximately 6 samples) are used for testing. From the other 99 subset speakers, 50 samples are also randomly selected to be used as impostor trials.
- Subset 2. Consisting of 100 speakers that will be used as CTS. One random sample (file) is selected from each speaker. Then, the size of the CTS is 100 samples, with an equalized composition in channel types.
- Subset 3. Consisting of 100 speakers that will be used to perform ZNorm. One random sample (file) is selected from each speaker. Then, the size of this subset is 100 samples, also with an equalized composition in channel types .

To get conclusive results about the final system configuration, some of the tests shown in the next section were performed with four different selections of the previous subsets.

4.2 System Parameter Selection

Different experiments were performed to test the system performance with regard to (some of these experiments were performed to confirm previous results with the new data):

- The number of MLP training epochs. Tests were performed with 50, 75, 100 and 150 MLP training epochs.
- The number of NTIL iterations. The results with 1, 2, 3, 4 and 5 NTIL iterations were achieved. The value of N in the NTIL algorithm was fixed to get approximately twice the quantity of vectors to those in the true speaker training set. That is, if, for example, the number of target speaker C training vectors is 5,000, for each NTIL iteration, we choose the N samples of the CTS with the highest scores, such that the sum of the vectors of these N samples will be approximately equal to 10,000.
- Calculate the final system score with and without the application of the R262 rule.
- The channel compensation scheme used. Two different parameter extractions were tested:

- **Parameters1.** Standard feature extraction is accomplished from each speech sample through 19 mel-frequency cepstral coefficients (MFCC), plus 19 ΔMFCC vectors. These are extracted from 32 ms frames, taken every 16 ms with Hamming windowing. A pre-emphasis factor of 0.97 is accomplished, and Cepstral Mean Subtraction (CMS) is applied as the channel compensation scheme.
- **Parameters2.** This is the same as the above, but the RASTA technique is also applied to compensate the channel influence.

– Train the MLP with and without normalization with regard to the true training sample channel type. Where we call:

- **Channel Type Normalized Training (CTNT)**, if the channel type ("land" and "cell" in the development data) of the TS is the same as that of the true speaker training sample.
- **Channel Type Non-Normalized Training (CTNNT)**, if the channel type of the true speaker training sample is not taken into account to select the TS.

The first three items show experiments performed to confirm previous results. Only the conclusions of these experiments are shown, for the sake of clarity.

With regard to the number of MLP training epochs, the results showed that 75 epochs are enough. The use of fewer epochs results in a worse performance, and the results are not better if more epochs are used and, in addition, the training time is increased.

As in previous works, the number of optimal NTIL algorithm iterations is 4, although the results are only a little better than those achieved with 3 iterations.

The advantages of using the R262 rule were also confirmed. Although the performance improvement is not very big, between 5% and 15%, the simplicity of the rule makes it interesting.

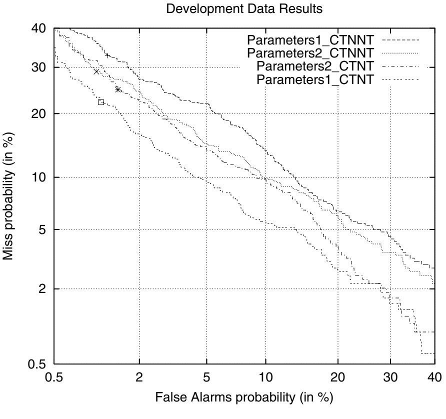


Fig. 1. Results achieved with the different system configurations tested with the development data

Table 1. System performance with the different system configurations tested with the development data, measured by means of DCF and EER, both in %

	Par1 and CTNT	Par1 and CTNNT	Par2 and CTNT	Par2 and CTNNT
DCF	9.1	14.8	11.4	12.0
EER	7.4	11.7	9.7	10.0

In fig. 1, the system performance can be seen with regard to the different alternatives shown in the last two items of the previous list. The performance of each system configuration, now measured by means of optimal Detection Cost Function (DCF) and Equal Error Rate (EER), is shown in table 1. The DCF is calculated as shown in the ISCSLP evaluation plan:

$$C_{DET} = C_{Miss} \times P_{Miss} \times P_{Target} + C_{FalseAlarm} \times P_{FalseAlarm} \times (1 - P_{Target}) \quad (5)$$

Where, $C_{Miss} = 10$, $C_{FalseAlarm} = 1$ and $P_{Target} = 0.05$.

From the results, it is easy to see that the best system configuration is with *parameters1* and CTNT. But a new problem appears: the channel type is available in the development data, but is not available in the evaluation data. Then, a channel classifier is necessary. This will be described in the next section, where the influence of channel misclassification in the system performance is also shown.

4.3 Channel Type Classifier

The channel type classification is also a two class classification problem, since the channel types in the development set are two: “land” and “cell”. Then, the same classifier, a discriminative MLP with the architecture shown in section 3, was used.

The speech feature extraction is the same as that shown in the previous section, but without channel compensation.

To train and test the classifier, the 300 speakers included in the development data were then split into the following two randomly selected different subsets:

- Subset 1. Consisting of 250 speakers. One sample (file) with “land” channel and another with “cell” channel were randomly selected for each speaker to train the classifier.
- Subset 2. Consisting of 50 speakers, whose samples are used for testing.

The advantages of using CTNT are clear from the results, but what happens if the sample is misclassified? Table 2 shows the high degradation of the system performance if the true speaker training sample channel type and the TS channel type are different.

As not enough data were available to get conclusive results, the following was done to decrease the error probability in the final decision:

Table 2. System performance when CTNT is used and when the true speaker training sample channel type and the TS channel type are different (column *miss* in the table). *Parameters 1* are used.

	CTNT	miss
DCF	9.1	19.8
EER	7.4	18.1

1. Over 40 different classifiers were trained with different selections of the previous subsets.
2. The four with the best performance were chosen. The EER achieved in these classifiers is about 2%-3%.
3. The final decision was based on the individual decision of each of the four selected classifiers: only if three or all four classifiers classify a sample as "cell" or "land" is this sample then finally classified as "cell" or "land", respectively; otherwise the final decision is "none". What happens in this situation is shown in the next section. The decision threshold was fixed at the equal error point of each classifier.

5 Final Systems. Evaluation Results

Although the best system configuration is clear, it was decided to use two different systems, due to the uncertainty concerning the channel classifier performance:

- **System 1.** If the channel classifier final decision is "cell" or "land", then the speech feature extraction is that shown in *parameters1*, and CTNT is used. If the decision of the channel classifier is "none" the system configuration is that shown in *System 2.*
- **System 2.** If the channel misclassification in the evaluation data was high, then the system performance would be bad. Then, CTNNT is used in this system to avoid this problem. The feature extraction is that shown in *parameters2*, since its performance with CTNNT is better than that with *parameters1* (see fig. 1).

The same CTS used in the last experiment performed with the development data was used as CTS in the evaluation.

Fig. 2 shows the performance of both systems by means of a DET curve. The optimum DCF achieved is 10.91% with *System 1* and 15.25% with *System 2.* Although comparison with other participant results cannot be done, it can be said that *System 1* is the third best system. Although it is far from the first, it is very close to the second.

The real DCF achieved is 10.8% with *System 1* and 15.4% with *System 2.* These values are very similar to the optimal ones.

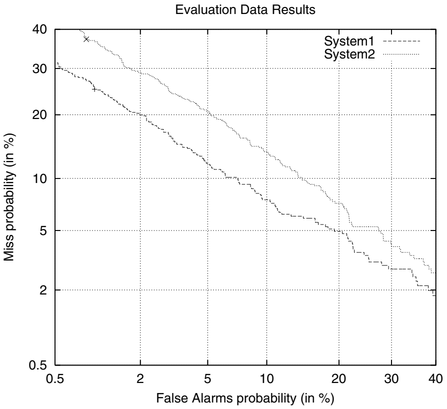


Fig. 2. Results achieved with the evaluation data

6 Conclusions

In this work not only the SV task has been approached, but also the channel type classification task.

With respect to this second task, the difference between system 1 results and that achieved with the evaluation data, where the channel type is known, is not very high. Thus, it can be concluded that although the channel type classifier proposed can be improved, it has a good performance. We believe it would be easy to improve this system, since, due to the time limitation, the system used is the first proposed, and other configurations or systems could not be tested. Besides, the evaluation data is also available to improve the system.

With respect to the SV task, two systems have been proposed, the best being that which uses CTNT (*System 1*). Although the results are not the best, from the *System 1* performance, it can be concluded that, in spite of the lack of attention to the use of discriminative ANNs in the SV task, this kind of classifier can be an interesting alternative.

Finally, we believe that another interesting result is that the optimum DCF and the real DCF are very close. This implies that the behaviour of the systems proposed is predictable, which is an interesting property for getting real SV systems.

References

1. T. Artieres, Y. Bennani, P. Gallinari, and C. Montacie. Connectionist and conventional models for free text talker identification. In *Proc. Neuro-Nimes, France*, 1991.
2. Gustavo Batista. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD explorations*, 6(1):20–29, 2004.

3. M. Bianchini, P. Frasconi, and M. Gori. Learning in multilayered networks used as autoassociators. *IEEE Trans. on Neural Networks*, 6(2):512–515, March 1995.
4. Nitesh V. Chawla, Nathalie Japkowicz, and Aleksander Kotcz. Editorial: Special issue on learning from imbalance data sets. *SIGKDD explorations*, 6(1):1–6, 2004.
5. Assaleh K. T. Farrel K. R., Mammone R. J. Speaker recognition using neural networks and conventional classifiers. *IEEE Transactions on Speech and Audio Processing*, 2, No. 1, part II, 1994.
6. Todor Ganchev, Dimitris Tasoulis, Michael N. Vrahatis, and NIkos Fakotakis. Locally recurrent probabilistic neural network for text-independent speaker verification. In *Proc. Eurospeech 03*, pages 1673–1676, 2003.
7. Piotr Juszczak and Robert P.W. Duin. Uncertainty sampling methods for one-class classifiers. In *Proc. of the Workshop on Learning from Imbalanced Datasets II, ICML*, 2003.
8. Mark A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE*, 37(2):233–243, February 1991.
9. Itshak Lapidot. Som as likelihood estimator for speaker clustering. In *Proc. Eurospeech 03*, pages 3001–3004, 2003.
10. Steve Lawrence, Ian Burns, Andrew Back, Ah Chung Tsoi, and C. Lee Giles. Neural networks classification and prior class probabilities. *Lecture Notes in Computer Science*, pages 299–314, 1998.
11. A. J. Mansfield and J. L. Wayman. Best pratices in testing and reporting performance of biometric devices. version 2.01. Technical report, 2002.
12. Leena Mary, K. Sri Rama Murty, S.R. Mahadeva Prasanna, and B. Yegnanarayana. Features for speaker and language identification. In *Proc. Odyssey 2004, the Speaker and Language Recognition Workshop*, 31 May-3 June 2004.
13. J. Oglesby and J. S. Mason. Optimization of neural models for speaker identification. In *Proceedings IEEE ICASSP*, volume S5-1, pages 261–264, 1990.
14. J. Oglesby and J.S. Mason. Radial basis function networks for speaker recognition. In *Proc. IEEE ICASSP*, volume S6.7, pages 393–396. IEEE, 1991.
15. Carlos E. Vivaracho, Javier Ortega-Garcia, Luis Alonso, and Quiliano I. Moro. Extracting the most discriminant subset from a pool of candidates to optimize discriminant classifier training. *Lecture Notes in Computer Science/Lecture Notes in Artificial Intelligence, Foundations of Intelligent Systems, 14th Iternational Symposium ISMIS*, (2871):640–645, 2003.
16. Carlos E. Vivaracho, Javier Ortega-Garcia, Luis Alonso, and Quiliano I. Moro. Improving the competitiveness of discriminant neural networks in speaker verification. In *Proc. Eurospeech, ISSN 1018-4074*, pages 2637–2640, september 2003.
17. C. Vivaracho-Pascual, J. Ortega-Garcia, L. Alonso-Romero, and Q. Moro-Sancho. A comparative study of mlp-based artificial neural networks in text-independent speaker verification against gmm-based systems. In Borge Lindberg Paul Dalsgaard and Henrik Benner, editors, *Proc. of Eurospeech01*, volume 3, pages 1753–1756. ISCA, 3-7 September 2001.
18. Vicent Wan and Steve Renals. Speaker recognition using sequence discriminant support vector machines. *IEEE Transactions on Speech and Audio Processing*, 13(2):203–210, 2005.
19. B. Yegnanarayana and S.P. Kishore. Aann: an alternative to gmm for pattern recognition. *Neural Networks*, 15(3):459–469, April 2002.

Evaluation of EMD-Based Speaker Recognition Using ISCSLP2006 Chinese Speaker Recognition Evaluation Corpus

Shingo Kuroiwa[1], Satoru Tsuge[1], Masahiko Kita[1], and Fuji Ren[1,2]

[1] Faculty of Engineering, The University of Tokushima,
Tokushimashi 770-8506, Japan
{kuroiwa, tsuge, ren}@is.tokushima-u.ac.jp
http://a1-www.is.tokushima-u.ac.jp/
[2] School of Information Engineering, Beijing University of Posts and
Telecommunications Beijing 100876

Abstract. In this paper, we present the evaluation results of our proposed text-independent speaker recognition method based on the Earth Mover's Distance (EMD) using *ISCSLP2006 Chinese speaker recognition evaluation* corpus developed by the Chinese Corpus Consortium (CCC). The EMD based speaker recognition (EMD-SR) was originally designed to apply to a distributed speaker identification system, in which the feature vectors are compressed by vector quantization at a terminal and sent to a server that executes a pattern matching process. In this structure, we had to train speaker models using quantized data, so that we utilized a non-parametric speaker model and EMD. From the experimental results on a Japanese speech corpus, EMD-SR showed higher robustness to the quantized data than the conventional GMM technique. Moreover, it has achieved higher accuracy than the GMM even if the data were not quantized. Hence, we have taken the challenge of *ISCSLP2006 speaker recognition evaluation* by using EMD-SR. Since the identification tasks defined in the evaluation were on an open-set basis, we introduce a new speaker verification module in this paper. Evaluation results showed that EMD-SR achieves 99.3% *Identification Correctness Rate* in a closed-channel speaker identification task.

Keywords: speaker identification, Earth Mover's Distance, nonparametric, vector quantization, Chinese speech corpus.

1 Introduction

In recent years, the use of portable terminals, such as cellular phones and PDAs (Personal Digital Assistants), has become increasingly popular. Additionally, it is expected that almost all appliances will connect to the Internet in the future. As a result, it will become increasingly popular to control these appliances using mobile and hand-held devices. We believe that a speaker recognition system will be used as a convenient personal identification system in this case.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 539–548, 2006.

In order to meet this demand, we have proposed some speaker recognition techniques[1,2,3] that focused on Distributed Speech/Speaker Recognition (DSR) systems[4,5,6,7,8,9,10]. DSR separates the structural and computational components of recognition into two components - the front-end processing on the terminal and the matching block of the speech/speaker recognition on the server. One advantage of DSR is that it can avoid the negative effects of a speech codec because the terminal sends the server not a compressed speech signal but quantized feature parameters. Therefore, DSR can lead to an improvement in recognition performance. In speech recognition services, DSR is widely deployed in Japanese cellular telephone networks[11]. On the other hand, in speaker recognition, since a speaker model has to be trained with a little registration voice, quantization poses a big problem, especially in case of using a continuation probability density function e.g. GMM[9,10].

To solve this problem, we propsed a non-parametric speaker recogniton method that did not require estimating statistical parameters of the speaker model[2]. We represented a speaker model using a histogram of speaker-dependent VQ codebooks (VQ histogram). To calculate the distance between the speaker model and the feature vectors for recognition, we applied the Earth Mover's Distance (EMD) algorithm. The EMD algorithm has been applied to calculate the distance between two images represented by histograms [1] of multidimensional features[12]. In [2], we conducted text-independent speaker identification experiments using the Japanese de facto standard speaker recognition corpus and obtained better performance than the GMM for quantized data. After that, we extended the algorithm to calculate the distance between a VQ histogram and a data set. From the results, we observed it achieved higher accuracy than the GMM and VQ distortion method even if the data were not quantized. We consider that the better results were obtained by the proposed method, since it can compare the distribution of the speaker model with the distribution of the testing feature vectors as is.

To evaluate the proposed method using a larger database, we have taken the challenge of the speaker recognition evaluation organized by the Chinese Corpus Consortium (CCC) for *the 5th International Symposium on Chinese Spoken Language Processing* (*ISCSLP 2006*). In view of the characteristics of the proposed method, we have chosen the text-independent speaker recognition task from the five tasks provided by CCC. The method was originally designed for the classical speaker identification problem that does not require a function to reject out-of-set speaker voices. However, since the evaluation data includes out-of-set speaker voices, we introduce a new speaker verification module in this paper.

This paper will continue as follows. Section **2** explains the Earth Mover's Distance and the originally proposed speaker identification method. Some modifications for *ISCSLP2006 speaker recognition evaluation* are also described. Section **3**

[1] In [12], EMD is defined the distance between two *signatures*. The *signatures* are histograms that have different bins, so that we use "histogram" as a term in this paper.

presents speaker identification experiments using *ISCSLP2006 speaker recognition evaluation* corpus. Finally, we summarize this paper in section **4**.

2 Nonparametric Speaker Recognition Method Using EMD

In this section, we first provide a brief overview of Earth Mover's Distance. Next, we describe the distributed speaker recognition method using a nonparametric speaker model and EMD measurement. Finally, we propose EMD speaker identification for unquantized data and a speaker verification module for identifing out-of-set speaker voices.

2.1 Earth Mover's Distance

The EMD was proposed by *Rubner et al.*[12] for an efficient image retrieval method. In this section, we describe the EMD algorithm.

The EMD is defined as the minimum amount of work needed to transport *goods* from several *suppliers* to several *consumers*. The EMD computation has been formalized by the following linear programming problem: Let $\boldsymbol{P} = \{(\boldsymbol{p}_1, w_{p_1}), \ldots, (\boldsymbol{p}_m, w_{p_m})\}$ be the discrete distribution, such as a histogram, where $\boldsymbol{p}_i$ is the centroid of each cluster and w_{p_i} is the corresponding weight (=frequency) of the cluster; let $\boldsymbol{Q} = \{(\boldsymbol{q}_1, w_{q_1}), \ldots, (\boldsymbol{q}_n, w_{q_n})\}$ be the histogram of test feature vectors: and $\boldsymbol{D} = [d_{ij}]$ be the ground distance matrix where d_{ij} is the ground distance between centroids $\boldsymbol{p}_i$ and $\boldsymbol{q}_j$.

We want to find a flow $\boldsymbol{F} = [f_{ij}]$, where f_{ij} is the flow between $\boldsymbol{p}_i$ and $\boldsymbol{q}_j$ (i.e. the number of *goods* sent from $\boldsymbol{p}_i$ to $\boldsymbol{q}_j$), that minimizes the overall cost

$$WORK(\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{F}) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}, \tag{1}$$

subject to the following constraints:

$$f_{ij} \geq 0 \qquad (1 \leq i \leq m, 1 \leq j \leq n), \tag{2}$$

$$\sum_{j=1}^{n} f_{ij} \leq w_{p_i} \qquad (1 \leq i \leq m), \tag{3}$$

$$\sum_{i=1}^{m} f_{ij} \leq w_{q_j} \qquad (1 \leq j \leq n), \tag{4}$$

$$\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} = \min \left(\sum_{i=1}^{m} w_{p_i}, \sum_{j=1}^{n} w_{q_j} \right). \tag{5}$$

Constraint (2) allows moving *goods* from $\boldsymbol{P}$ to $\boldsymbol{Q}$ and not vice versa. Constraint (3) limits the amount of *goods* that can be sent by the cluster in $\boldsymbol{P}$ to their weights. Constraint (4) limits the amount of *goods* that can be received by the cluster in $\boldsymbol{Q}$ to their weights. Constraint (5) forces movement of the maximum amount of *goods* possible. They call this amount the *total flow.* Once the transportation problem is solved, and we have found the optimal flow $\boldsymbol{F}$, the EMD is defined as the work normalized by the total flow:

$$EMD(\boldsymbol{P},\boldsymbol{Q}) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} f_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}} \tag{6}$$

The normalization factor is the total weight of a smaller distribution, because of constraint (5). This factor is needed when the two distributions of *suppliers* have different total weight, in order to avoid favoring a smaller distribution.

2.2 The Recognition Flow of the Proposed Method

In the previous section, we described that EMD is calculated as the least amount of work which fills the requests of *consumers* with the goods of *suppliers.*

If we define the speaker model as the *suppliers* and the testing feature vectors as the *consumers*, the EMD can be applied to speaker recognition. Hence, we propose a distributed speaker recognition method using a nonparametric speaker model and EMD measurement. The proposed method represents the speaker model and testing feature vectors as a histogram. The detail of the proposed method is described as follows.

Fig. 1 illustrates a flow of the feature extraction process using the ETSI DSR standard[5] and the proposed method. In the figure, dotted ($\dot{}$) elements indicate data quantized once and double dotted ($\ddot{}$) elements indicate data quantized twice. As shown in the upper part of the figure, both registered utterances and testing utterances are converted to quantized feature vector sequences, $\dot{\boldsymbol{V}}_A, \dot{\boldsymbol{V}}_B, \ldots$, and $\dot{\boldsymbol{V}}_X$, using the ETSI DSR front-end and back-end (N_A, N_B, and N_X are the number of frames in each sequence). In this block, $\boldsymbol{c}_t$ is a feature vector of time frame t that consists of MFCC and logarithmic energy; $\boldsymbol{x}_t$ is a code vector that is sent to the back-end (server); $\dot{\boldsymbol{c}}_t$ is a decompressed feature vector; and $\dot{\boldsymbol{v}}_t$ is a feature vector for use in the subsequent speaker recognition process. Using $\dot{\boldsymbol{V}}_A, \dot{\boldsymbol{V}}_B, \ldots$, and $\dot{\boldsymbol{V}}_X$, the proposed method is executed as follows.

(a) Speaker Model Generation. Using the registered feature vectors, the system generates each speaker's VQ codebook, $\{\ddot{\boldsymbol{p}}_1^{sp}, \ldots, \ddot{\boldsymbol{p}}_m^{sp}\}$, by using the LBG algorithm with Euclidean distance, where sp is a speaker name, and m is a codebook size. In order to make a histogram of VQ centroids, the number of registered vectors whose nearest centroid is $\ddot{\boldsymbol{p}}_i^{sp}$ is counted and frequency is set to $w_{p_i}^{sp}$. As a result, we get a histogoram of the speaker, sp, that is the speaker model in the proposed method,

$$\boldsymbol{P}^{sp} = \{(\ddot{\boldsymbol{p}}_1^{sp}, w_{p_1}^{sp}) \ldots, (\ddot{\boldsymbol{p}}_m^{sp}, w_{p_m}^{sp})\}. \tag{7}$$

This histogram is used as the *suppliers*' discrete distribution, $\boldsymbol{P}$, described in the previous section.

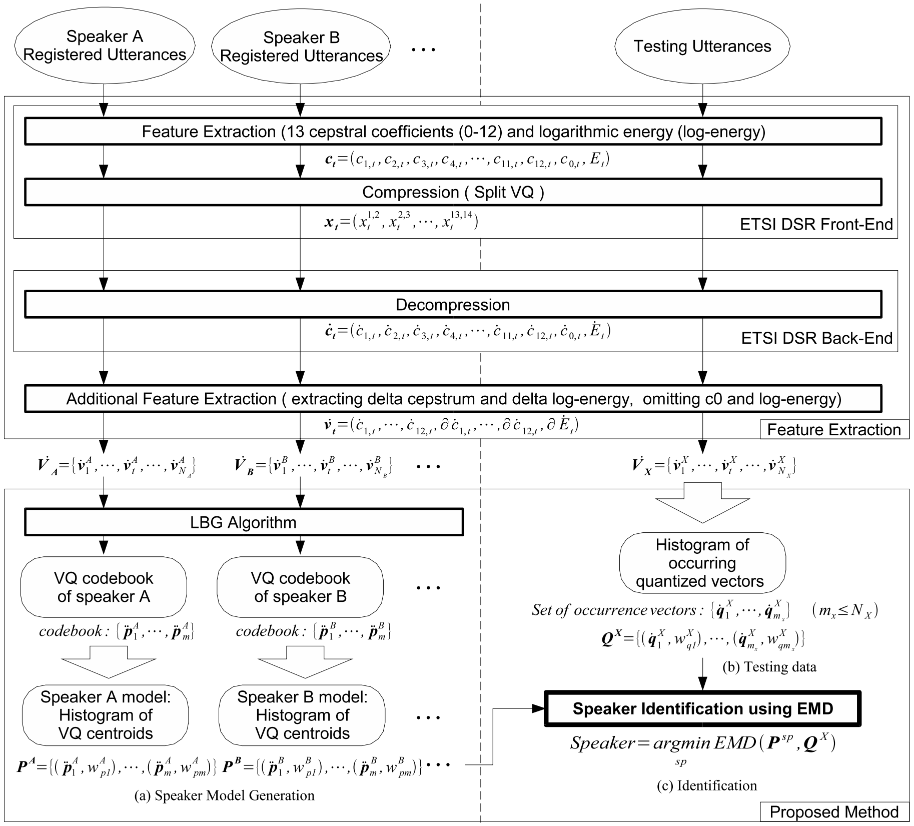

Speaker A Registered Utterances
Speaker B Registered Utterances
Testing Utterances
Feature Extraction (13 cepstral coefficients (0-12) and logarithmic energy (log-energy)
$\boldsymbol{c_t}=(c_{1,t}, c_{2,t}, c_{3,t}, c_{4,t}, \cdots, c_{11,t}, c_{12,t}, c_{0,t}, E_t)$
Compression (Split VQ)
$\boldsymbol{x_t}=(x_t^{1,2}, x_t^{2,3}, \cdots, x_t^{13,14})$
ETSI DSR Front-End
Decompression
$\dot{\boldsymbol{c}}_t=(\dot{c}_{1,t}, \dot{c}_{2,t}, \dot{c}_{3,t}, \dot{c}_{4,t}, \cdots, \dot{c}_{11,t}, \dot{c}_{12,t}, \dot{c}_{0,t}, \dot{E}_t)$
ETSI DSR Back-End
Additional Feature Extraction (extracting delta cepstrum and delta log-energy, omitting c0 and log-energy)
$\dot{\boldsymbol{v}}_t=(\dot{c}_{1,t}, \cdots, \dot{c}_{12,t}, \partial\dot{c}_{1,t}, \cdots, \partial\dot{c}_{12,t}, \partial\dot{E}_t)$
Feature Extraction
$\dot{\boldsymbol{V}}_{\boldsymbol{A}}=\{\dot{\boldsymbol{v}}_1^A, \cdots, \dot{\boldsymbol{v}}_t^A, \cdots, \dot{\boldsymbol{v}}_{N_A}^A\}$
$\dot{\boldsymbol{V}}_{\boldsymbol{B}}=\{\dot{\boldsymbol{v}}_1^B, \cdots, \dot{\boldsymbol{v}}_t^B, \cdots, \dot{\boldsymbol{v}}_{N_B}^B\}$
$\dot{\boldsymbol{V}}_{\boldsymbol{X}}=\{\dot{\boldsymbol{v}}_1^X, \cdots, \dot{\boldsymbol{v}}_t^X, \cdots, \dot{\boldsymbol{v}}_{N_X}^X\}$
LBG Algorithm
VQ codebook of speaker A
VQ codebook of speaker B
$codebook: \{\ddot{\boldsymbol{p}}_1^A, \cdots, \ddot{\boldsymbol{p}}_m^A\}$
$codebook: \{\ddot{\boldsymbol{p}}_1^B, \cdots, \ddot{\boldsymbol{p}}_m^B\}$
Speaker A model: Histogram of VQ centroids
Speaker B model: Histogram of VQ centroids
$\boldsymbol{P^A}=\{(\ddot{\boldsymbol{p}}_1^A, w_{p1}^A), \cdots, (\ddot{\boldsymbol{p}}_m^A, w_{pm}^A)\}$
$\boldsymbol{P^B}=\{(\ddot{\boldsymbol{p}}_1^B, w_{p1}^B), \cdots, (\ddot{\boldsymbol{p}}_m^B, w_{pm}^B)\}$
(a) Speaker Model Generation
Histogram of occurring quantized vectors
$Set\ of\ occurrence\ vectors: \{\dot{\boldsymbol{q}}_1^X, \cdots, \dot{\boldsymbol{q}}_{m_x}^X\} \quad (m_x \leq N_X)$
$\boldsymbol{Q^X}=\{(\dot{\boldsymbol{q}}_1^X, w_{q1}^X), \cdots, (\dot{\boldsymbol{q}}_{m_x}^X, w_{qm_x}^X)\}$
(b) Testing data
Speaker Identification using EMD
$Speaker = \underset{sp}{argmin}\, EMD(\boldsymbol{P}^{sp}, \boldsymbol{Q}^X)$
(c) Identification
Proposed Method

Fig. 1. A block diagram of the feature extraction process and the proposed speaker recognition method

(b) Testing data. A histogram of testing data is directly calculated from $\dot{\boldsymbol{V}}_X$ that was quantized by ETSI DSR standard. The quantized feature vectors consist of static cepstrum vectors that have 64^6 possible combinations and their delta cepstrum vectors, so that we consider they are a set of vectors, $\{\dot{\boldsymbol{q}}_1^X, \ldots, \dot{\boldsymbol{q}}_{m_x}^X\}$, where m_x is the number of individual vectors. In order to make a histogram of the set of vectors, occurrence frequency of the vector $\dot{\boldsymbol{q}}_i^X$ is set to $w_{q_i}^X$. As a result, we get a histogoram of the testing data,

$$\boldsymbol{Q}^X = \{(\dot{\boldsymbol{q}}_1^X, w_{q_1}^X) \ldots, (\dot{\boldsymbol{q}}_{m_x}^X, w_{q_{m_x}}^X)\}. \tag{8}$$

This histogram is used as the *consumers*' discrete distribution, $\boldsymbol{Q}$, described in the previous section.

(c) Identification. Using the speaker models, $\boldsymbol{P}^{sp}$, and the testing data, $\boldsymbol{Q}^X$, speaker recognition is executed as in the following equation.

$$Speaker = \underset{sp}{argmin}\, EMD(\boldsymbol{P}^{sp}, \boldsymbol{Q}^X) \tag{9}$$

As the grand distance, d_{ij}, in EMD, we use the Euclidean distance between $\ddot{\boldsymbol{p}}_i^{sp}$ and $\dot{\boldsymbol{q}}_j^X$. Since we utilize $w_{p_i}^{sp}$ and $w_{q_j}^X$ as the frequency of $\ddot{\boldsymbol{p}}_i^{sp}$ and $\dot{\boldsymbol{q}}_j^X$ respectively, f_{ij} is the number of vectors matched with $\ddot{\boldsymbol{p}}_i^{sp}$ and $\dot{\boldsymbol{q}}_j^X$ (i.e. the number of *goods* sent from $\ddot{\boldsymbol{p}}_i^{sp}$ to $\dot{\boldsymbol{q}}_j^X$), that minimizes the overall cost by EMD.

2.3 Modification to Adapt to Unquantized and Out-of-Set Data

In order to apply the proposed method to unquantized data and to identify out-of-set data, i.e. *ISCSLP2006 speaker recognition evaluation* corpus, we have modified the recognition flow described in the previous section.

First, the "Compression" and "Decompression" blocks in Fig. 1 are skipped and consequently feature vector sequences, $\dot{\boldsymbol{V}}_A, \dot{\boldsymbol{V}}_B, \ldots$, and $\dot{\boldsymbol{V}}_X$, become unquantized feature vector sequences, $\boldsymbol{V}_A, \boldsymbol{V}_B, \ldots$, and $\boldsymbol{V}_X$. In "Speaker Model Generation", the LBG algorithm can generate each speaker's codebook from the unquantized feature vector sequence without any modification of the algorithm. On the other hand, we consider the set of the feature vectors of the testing utterance to be a histogram in which the occurrence frequency of each vector is one. By the above modification, we can calculate EMD between the speaker model and the testing data.

Next, we have introduced an out-of-set identification module after "Speaker identification using EMD" in Fig. 1. Although a candidate speakers list of each testing datum is provided in the evaluation, we calculate EMDs between the testing datum and all speaker models. After that, a confidence score of the nearest speaker in the candidate list is calculated by the following equation,

$$score(s) = -(EMD(\boldsymbol{P}^s, \boldsymbol{Q}^X) - \min_{sp \neq s} EMD(\boldsymbol{P}^{sp}, \boldsymbol{Q}^X)) + bias(s), \tag{10}$$

where s is the nearest speaker in the candidate list, $\boldsymbol{P}^{sp}$ is each of the speaker model in a speaker recognition system, $\boldsymbol{Q}^X$ is the testing datum and $bias(s)$ is speaker-dependent bias to control False Rejection Rate (FRR) and False Acceptance Rate (FAR). If the score is equal to or greater than zero, it is recognized as the testing data having been uttered by speaker s. If not, it is identified as the out-of-set datum. In the experiments we used 400 speaker models that were trained with all data for enrollment in the text-independent speaker recognition task of *ISCSLP2006 speaker recognition evaluation.* The $bias(s)$ is calculated by the follwong equations,

$$X_s = \underset{X \in testset}{argmin}\, EMD(\boldsymbol{P}^s, \boldsymbol{Q}^X), \tag{11}$$

$$bias(s) = EMD(\boldsymbol{P}^s, \boldsymbol{Q}^{X_s}) - \min_{sp \neq s} EMD(\boldsymbol{P}^{sp}, \boldsymbol{Q}^{X_s}) + globalbias, \tag{12}$$

where *globalbias* is one global value that controls the ratio of results for in-set and out-of-set. We assumed that at least one speaker's datum was contained in the testset, and $bias(s)$ is determined based on the score of this datum, i.e. the nearest datum in the testset. We think this is a reasonable way, because in a real application system, we can get each speaker's utterances that are uttered to access the system and we can know the ratio of users for in-set and out-of-set in a field trial phase of the system. Actually, we have a good example of this technique, that is, the threshold values in the Prank Call Rejection System[13] that was deployed by KDDI international telephone service from 1996 were determined with this kind of process and still works effectively today.

3 Experiments

We conducted text-independent speaker identification experiments to evaluate the proposed method using the *ISCSLP2006 spaker recognition evaluation* data developed by Chinese Corpus Consortium (CCC).

3.1 Task Definition

In *ICSLP2006*, CCC has organized a special session on speaker recognition and provided speech data to evaluate speaker recognition algorithms using the same database. CCC provided several kinds of tasks, i.e, text-independent speaker identification, text-dependent and text-independent speaker verification, text-independent cross-channel speaker identification, and text-dependent and text-independent cross-channel speaker verification. We chose the text-independent speaker recognition task in view of the characteristics of the proposed method. The data set of this task contained 400 speakers' data for enrollment, and 2,395 utterances for testing. Each datum to enroll is longer than 30 seconds and recorded over PSTN or GSM network. In testing data, the enrolled speaker uttered over same kind of channels in this task. Each testing datum has a candidate speakers list and about half of the testing data were uttered by out-of-set speakers who did not appear in the list. Therefore, the speaker identification algorithm has to decide whether each testing datum is in-set or out-of-set also.

CCC also provided the development data that contained 300 speakers' utterances with speaker labels and channel conditions. We were able to decide the various parameters of the algorithm using that data.

The performance of speaker identification was evaluated by *Identification Correctness Rate*, defined as:

$$\%CorrectIdentification = \frac{NumberOfCorrectlyIdentifiedData}{TotalNumberOfTrialData} \times 100\%, \tag{13}$$

where "correctly identified data" means those data identified as the speaker models they shoud be by the top-candidate output, if they are "in-set", or "non-match" if "out-of-set".

3.2 Experimental Conditions

All data, sampled at 8kHz, were segmented into overlapping frames of 25ms, producing a frame every 10ms. A Hamming window was applied to each frame. Mel-filtering was performed to extract 12-dimensional static MFCC, as well as a

Table 1. Evaluation results with the proposed method

Identification Correctness Rate	99.33 %	(2379/2395)
False Acceptance Rate	0.42 %	(10/2395)
False Rejection Rate	0.25 %	(6/2395)
Recognition Error Rate	0.00 %	(0/2395)

logarithmic energy (log-energy) measure. The 12-dimensional delta MFCC and delta log-energy were extracted from the static MFCC and the log-enrgy, respectively. After that, by omitting the log-energy, we constituted a 25-dimensional feature vector (12 static MFCCs + 12 delta MFCC + delta log-energy). Cepstral Mean Subtraction (CMS) was applied on the static MFCC vectors. We used HTK3.3[16] for the feature extraction.

In order to avoid an influence of non-speech sections and unreliable speech frames, we employed a voice activity detector (VAD) that classifies each frame into speech or background sound on a frame-by-frame basis. The VAD used power threshold that was calculated from percentile levels based on each observed speech signal. We used the following threshold in the experiments.

$$Threshold = (P_{95\%tile} - P_{10\%tile}) \times 0.2 + P_{10\%tile}, \tag{14}$$

Only the frames with higher power level than this threshold value were used for speaker identification. This process reduced the number of frames by 10% to 50%.

In the experiment, we set the number of centroids of each speaker's codebook to 64, which gave the best accuracy in experiments using the development data. The speaker dependent threshold for detecting the out-of-set data was also set up using this data and the previous information that the ratio of testing samples for in-set and out-of-set cases would be about 1:1.

3.3 Experimental Results

Table 1 shows the *Identification Correctness Rate* (ICR), False Acceptace Rate (FAR), False Rejection Rate (FRR), and Recognition Error Rate (RER). RER is the rate which identified the utterance of one speaker in the candidate list as another speaker's utterance. The table shows the proposed method achieved extremely high performance in the open-set manner. This result is the best ICR in the "speaker identification task" under the closed-channel condition of *ISCSLP2006 speaker recognition evaluation.* On the other hand, the proposed algorithm required much computation time. Actually, it took about four minutes to identify one utterance with an Intel Pentium 4 3.2GHz processor in the experiments.

When we investigated the data of FAR and FRR, the word sequences of several testing data were included in the training data of the other speaker and was not included in the training data of the correct speaker. The use of automatic speech recognition will improve the speaker identification performance for these data[1,14], although it will turn into a language dependent system.

Although we did not compare the proposed method with the conventional speaker identification techniques in this paper, we think it outperforma than GMM and VQ distortion techniques. Because the proposed method directly calculates the distance between data set, while both of the VQ distortion and GMM methods calculate the distance by totaling the VQ distortion or the likelihood of each frame. The proposed method can compare the distribution of the training feature vectors with the distribution of the testing feature vectors. Actually, we observed better performance with the proposed method in a Japanese corpus.

4 Summary

In this paper, we have presented evaluation results of a novel speaker recognition method based on a nonparametric speaker model and Earth Mover's Distance (EMD) using *ISCSLP2006 speaker recognition evaluation* corpus provided by the Chinese Corpus Consortium (CCC). The proposed method was originally designed to apply to a distributed speaker recognition system. We have improved the method to be able to handle unquantized data and reject out-of-set speakers in this paper.

Experimental results on the *ISCSLP2006* text-independent speaker identification task under the closed-channel condition in the open-set manner, showed that the proposed method achieved 99.33% *Identification Correctness Rate*, which is the best score in this task. This result suggests that the proposed method would be effective also in speaker verification. On the other hand, the proposed method needed much computation time. We also confirmed the errors of the proposed method depended on the contents of utterances.

In future work, we will accelerate the distance calculation process in the proposed algorithm and apply the method to speaker verification. Furthermore, we will consider use of speech recognition to improve the speaker identification accuracy.

Acknowledgments

This research has been partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Exploratory Research, 17656128, 2006, Japan Society for the Promotion of Science, Grant-in-Aid for Scientific Research (B), 17300065, 2006, and International Communications Foundation (ICF). We would also appriciate the Chinese Corpus Consortium that provided valuable speech data.

References

1. Mohamed Abdel Fattah, Fuji Ren, Shingo Kuroiwa, and Ippei Fukuda, “Phoneme Based Speaker Modeling to Improve Speaker Recognition,” Information, Vol.9, No.1, pp.135–147, Jan. 2006.

2. Shingo Kuroiwa, Yoshiyuki Umeda, Satoru Tsuge, and Fuji Ren, "Nonparametric Speaker Recognition Method using Earth Mover's Distance," IEICE Transactions on Information and Systems, Vol.E89-D, No.3, pp.1074–1081, Mar. 2006.
3. Mohamed Abdel Fattah, Fuji Ren, and Shingo Kuroiwa, "Effects of Phoneme Type and Frequency on Distributed Speaker Identification and Verification," IEICE Transactions on Information and Systems, Vol.E89-D, No.5, pp.1712–1719, May 2006.
4. D. Pearce, "Enabling New Speech Driven Services for Mobile Devices: An overview of the ETSI standards activities for Distributed Speech Recognition Front-ends," Proc. Applied Voice Input/Output Society Conference, Dec. 2000.
5. ETSI standard document, "Speech processing, transmission and auality aspects (STQ); Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithm," ETSI ES 201 108 v1.1.2, Apr. 2000.
6. http://www.itu.int/ITU-T/2001-2004/com16/sg16-q15.html
7. C. C. Broun, W. M. Campbell, D. Pearce, H. Kelleher, "Distributed Speaker Recognition Using the ETSI Distributed Speech Recognition Standard," Proc. A Speaker Odyssey - The Speaker Recognition Workshop, pp.121–124, June 2001.
8. S. Grassi, M. Ansorge, F. Pellandini, P.-A. Farine, "Distributed Speaker Recognition Using the ETSI AURORA Standard," Proc. 3rd COST 276 Workshop on Information and Knowledge Management for Integrated Media Communication, pp.120–125, Oct. 2002.
9. C.-H. Sit, M.-W. Mak, and S.-Y. Kung, "Maximum Likelihood and Maximum A Posteriori Adaptation for Distributed Speaker Recognition Systems," Proc. the 1st Int. Conf. on Biometric Authentication July, 2004.
10. I. Fukuda, M. A. Fattah, S. Tsuge, S. Kuroiwa, "Distributed Speaker Identification on Japanese Speech Corpus Using the ETSI Aurora Standard," Proc. 3rd Int. Conf. on Information, pp.207–210, Nov. 2004.
11. http://www.kddi.com/english/corporate/news_release/2006/0112/
12. Y. Rubner, L. Guibas, C. Tomasi, "The Earth Mover's Distance, Multi-Dimensional Scaling, and Color-Based Image Retrieval," Proc. the ARPA Image Understanding Workshop, pp.661–668, May 1997.
13. Shingo Kuroiwa, Shinichi Sakayori, Seiichi Yamamoto, Masanobu Fujioka: "Prank call rejection system for home country direct service", Proc. of IVTTA96, pp.135–138, Basking Ridge, U.S.A, Sep.-Oct. 1996.
14. A.Park and T.Hazen, "ASR dependent techniques for speaker identification", Proc. ICSLP2002, pp.1337–1340, Sep. 2002.
15. Toshiaki Uchibe, Shingo Kuroiwa, Norio Higuchi: "Determination of threshold for speaker verification using speaker adaptation gain in likelihood during training", Proc. ICSLP 2000, Vol. 2, pp.326–329, Beijing, China, Oct 2000.
16. http://htk.eng.cam.ac.uk/

Integrating Complementary Features with a Confidence Measure for Speaker Identification

Nengheng Zheng, P.C. Ching, Ning Wang, and Tan Lee

Department of Electronic Engineering, The Chinese University of Hong Kong,
Hong Kong
{nhzheng, pcching, nwang, tanlee}@ee.cuhk.edu.hk

Abstract. This paper investigates the effectiveness of integrating complementary acoustic features for improved speaker identification performance. The complementary contributions of two acoustic features, i.e. the conventional vocal tract related features MFCC and the recently proposed vocal source related features WOCOR, for speaker identification are studied. An integrating system, which performs a score level fusion of MFCC and WOCOR with a confidence measure as the weighting parameter, is proposed to take full advantage of the complementarity between the two features. The confidence measure is derived based on the speaker discrimination powers of MFCC and WOCOR in each individual identification trial so as to give more weight to the one with higher confidence in speaker discrimination. Experiments show that information fusion with such a confidence measure based varying weight outperforms that with a pre-trained fixed weight in speaker identification.

1 Introduction

Speaker recognition is an identity authentication process which automatically identifies individuals with the intrinsic characteristics conveyed by their voice. The state-of-the-art speaker recognition systems typically employ the Mel-frequency cepstral coefficients (MFCC) as the representative acoustic features [1], [2]. The MFCC features (with an appropriate pattern matching technique, e.g. Gaussian Mixture Model, GMM) have achieved very high recognition performances in laboratory conditions. However, the recognition accuracy degrades dramatically in real-world applications. Therefore, many efforts have been devoted to developing new acoustic features or high-level features to supplement the MFCC for improved speaker recognition performances [3, 4, 5, 6].

We have recently proposed a new feature, called the wavelet octave coefficients of residues (WOCOR), for speaker recognition [7]. As known, human speech can be modeled as the convolutional output of a vocal source excitation signal and the impulse response of the vocal tract filter system [8]. The MFCC parameters are derived mainly to represent the spectral envelope, or the formant structure of the vocal tract system [9]. The WOCOR parameters, on the other hand, aim at characterizing the spectro-temporal characteristics of the vocal source excitation [7]. These two features contain complementary speaker-specific information since they characterize two physiologically different information sources in speech

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 549–557, 2006.

production. The experiments presented in [7] have shown that a system combining MFCC and WOCOR achieves better performance than that using MFCC only in both speaker verification and identification.

The performance of the combining system, however, is highly relied on the effectiveness of the fusion technique selected. Generally, the information fusion can be done at (i) feature level, (ii) score level, or (iii) decision level. This paper proposes a score level fusion technique for combining MFCC and WOCOR for speaker identification. Score level fusion is preferred because the matching scores are easily obtained and contain sufficient information for distinguishing different speakers. In multimodal person authentication systems, which employ multiple biometric information to recognize a person, it is very important to apply efficient information fusion technique because the reliability of decision scores from different classifiers may vary significantly. There have been a number of works on developing information fusion technique for biometrics systems [10, 11, 12]. This paper proposes a confidence measure based fusion technique for the fusion of MFCC and WOCOR. The confidence measure is estimated from the matching scores, given by MFCC and WOCOR respectively, in each identification trial. The motivation for using such a confidence measure is that the distributions of matching scores show significant difference between the correct and incorrect identification trials. The confidence measure provides an optimized fusion score by giving more weight to the score with higher confidence to give a correct identification. Speaker identification experiments show that this system achieves better performance than that with a fixed weight scores fusion.

2 MFCC and WOCOR Feature Extraction

2.1 MFCC

The extraction of MFCC parameters follows the standard procedures as described in [9]:

1) Short-time Fourier transform is applied every 10 ms with a 30 ms Hamming window.

2) The magnitude spectrum is warped with a Mel-scale filter bank that consists of 26 filters, which emulates the frequency resolution of human auditory system. Log-magnitude of each filter output is calculated.

3) Discrete cosine transform (DCT) is applied to the filter bank output.

The MFCC feature vector has 39 components, including the first 12 cepstral coefficients, the log energy, as well as their first and second order time derivatives.

2.2 WOCOR

WOCOR parameters are generated based on pitch-synchronous wavelet transform of the linear predictive (LP) residual signal of voiced speech.

1) Detect the voiced speech and its pitch periods with the Robust Algorithm for Pitch Tracking (RAPT) algorithm [13]. Only voiced speech is kept for subsequent processing.

2) Generate the linear predictive (LP) residual signal by LP inverse filtering, i.e.,

$$e(n) = s(n) - \sum_{k=1}^{12} a_k s(n-k) \tag{1}$$

where the filter coefficients a_k are computed on every 30 ms. speech with Hamming window using the autocorrelation method [8]. To reduce the intra-speaker variation, the amplitude of $e(n)$ is normalized to the interval [-1, 1].

3) Pitch bursts of the residual signal are detected. The residual signal is divided into short segments with Hamming window. Each window covers exact two pitch cycles and synchronizes with the pitch bursts.

4) Apply wavelet transform on each Hamming windowed residual singal $e_h(n)$

$$w(a,b) = \frac{1}{\sqrt{|a|}} \sum_n e_h(n) \Psi^*(\frac{n-b}{a}) \tag{2}$$

where $\Psi(n)$ is the 4th-order Daubechies wavelet basis function, a and b are the scaling parameter and the translation parameter respectively [14]. Six scaling parameters $a = \{2^k | k = 1, 2 \cdots, 6\}$ is selected for multi-resolution analysis. As a result, there are six octave groups of wavelet coefficients, i.e.,

$$W_k = \left\{ w(2^k, b) \, | b = 1, 2, \cdots, N \right\}, k = 1, 2, ..., 6 \tag{3}$$

where N is the window length.

5) Each W_k is divided into 4 subgroups, i.e.,

$$\begin{aligned} &W_k^M(m) = \left\{ w(2^k, b) \, \middle| b \in (\tfrac{(m-1)N}{4}, \tfrac{mN}{4}] \right\} \\ &m = 1, \cdots, 4 \end{aligned} \tag{4}$$

The WOCOR feature vector with 24 parameters is generated as

$$\text{WOCOR} = \left\{ \|W_k(m)\| \, \middle| \begin{array}{l} m = 1, 2, \cdots, 4 \\ k = 1, 2, \cdots, 6 \end{array} \right\} \tag{5}$$

where $\| \cdot \|$ denotes the 2-norm operator.

To summarize, given a speech utterance, a sequence of WOCOR feature vectors are obtained by pitch-synchronous wavelet transform of the LP residual signal. Each feature vector consists of 24 components, which captures useful spectro-temporal characteristics of the residual signal.

3 Info-fusion with Confidence Measure

A simple and widely applied score fusion is the fixed weight linear combination, i.e.

$$s = w_t s_M + (1 - w_t) s_W \tag{6}$$

where s_M, s_W and s are the matching scores obtained from MFCC, WOCOR and the fused score, respectively. The fusion weight w_t can be pre-trained and is fixed the same for all the identification trials. A major problem of such a fixed weighting scheme is that it ignores the different contributions of MFCC and WOCOR in individual trials. For example, consider two types of identification trials as given in Table 1, in which MFCC and WOCOR give different contributions to speaker identification. With fixed weight fusion, though correct results are obtained for the Type I trials, incorrect decisions are made for Type II trials which have been correctly identified by MFCC only. To avoid this problem, an ideal solution should be capable of distinguishing the two cases and give null weight to MFCC for Type I trials and to null weight to WOCOR for Type II trials. Although such an ideal solution is not available in real-world applications, in the follows, we demonstrate that a confidence measure based fusion method with varying weights in different trials can avoid most of this kind of errors.

Table 1. Different contributions of MFCC and WOCOR in two types of trials

	Type I	Type II
MFCC	incorrect	correct
WOCOR	correct	incorrect
Fixed weight fusion	correct	incorrect

Analysis of the matching scores shows that, generally, in a correct identification, the difference of the scores between the identified speaker and the closest competitor is relatively larger than that in an incorrect identification. The score difference can therefore be adopted for measuring the discrimination power, i.e.

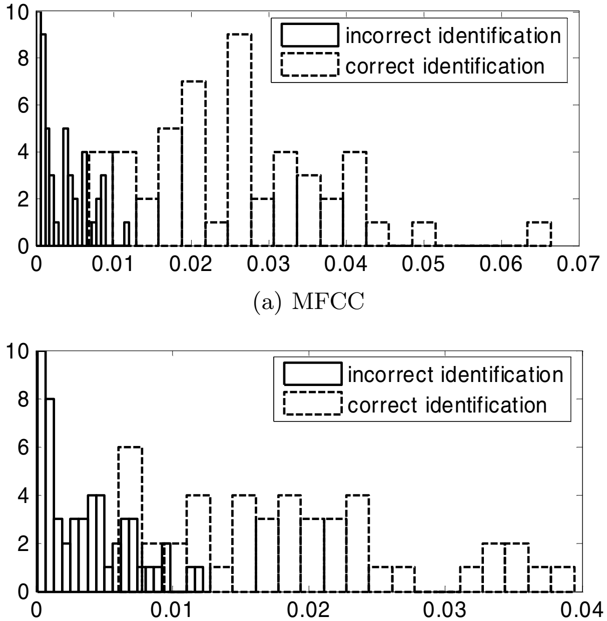
$$D = \frac{\max_i\{s_i\} - \mathit{second}\max_i\{s_i\}}{\left|\max_i\{s_i\}\right|} \tag{7}$$

where s_i is the matching score of the i-th speaker. The normalization of the difference over $\max\{s_i\}$ reduces the effect of the dynamic range of s.

Figure 1 shows the histogram of D of MFCC and WOCOR. It is clear that a correct identification is generally associated with a larger D than an incorrect identification. Therefore, a larger D implies that the corresponding feature has higher confidence for speaker identification. Obviously, it is preferable to taking into account D for score fusion in each identification trial instead of using the fixed weight. Although the optimal way to fuse the two scores with the knowledge of the discrimination power is not known, we found that a confidence measure (CM) including the discrimination ratio into the sigmoid function offers improvement to the identification performance. CM is then defined as

$$CM = -\log\frac{1}{1+e^{(-\alpha\cdot(DR-\beta))}} \tag{8}$$

where $DR = D_M/D_W$ is the discrimination ratio of MFCC and WOCOR, α and β control the slope of the mapping contour from DR to CM, as illustrated



(a) MFCC

(b) WOCOR

Fig. 1. Histogram of the speaker discrimination power D of the MFCC (a) and WOCOR (b) features

in Fig. 2. The solid line curve in Fig. 2 is used for the identification experiments described in Section 4. The corresponding parameters $\alpha = 0.2, \beta = -3$ are trained using the development data.

The score fusion with confidence measure for each identification trial is now written as

$$s = s_M + s_W \cdot CM \tag{9}$$

With CM, the fused score combines better weighted s_M and s_W based on the contributions of the corresponding features in that specific trial. As illustrated in Fig.2, when DR is large, CM tends to zero; the final decision will not be heavily affected by the score obtained from WOCOR. Contrarily, a small DR corresponds to a large CM, which will introduce more impact of score obtained from WOCOR for final decision.

4 Experiments

4.1 Experimental Setup

The UBM-GMM method [2] is adopted for training the speaker models. For each set of features, a universal background model (UBM) is first trained using the training data from all speakers. Then a Gaussian mixture model (GMM) is adapted from the UBM for each speaker using the respective training data.

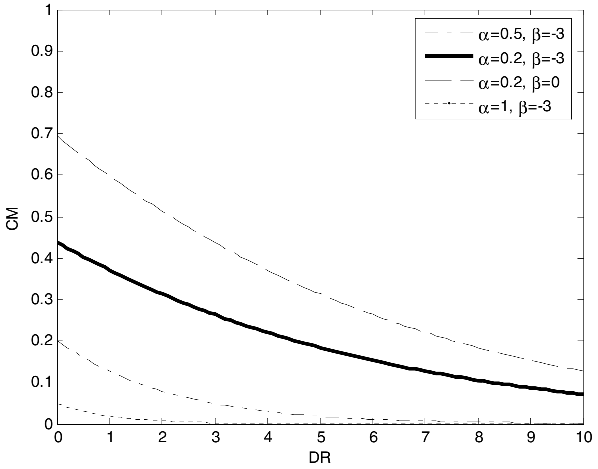


Fig. 2. Mapping contour from DR to CM

Although in real applications, the testing utterances could come from the unregistered impostors. In these experiments, we only deal with the close-set speaker identification. That is, all the testing utterances must come from one of the in-set speakers. The one whose GMM gives the highest matching score is marked as the identified speaker.

The speaker identification experiments are conducted on a subset of the CU2C database, which is a Cantonese speech database created in the Chinese University of Hong Kong for speaker recognition applications [15]. In the experiments, there are 50 male speakers, each having 18 sessions of speech data with 10 utterances in each session. The first 4 sessions are used for training the speaker models. Sessions 5-8 are used as development data for training the weighting parameters for the score level fusion of MFCC and WOCOR. Each speaker has 100 identification trials from the last 10 sessions. All the utterances are text-independent telephone speech with matched training and testing conditions (the same handset and fixed line telephone network). The speech data of each speaker are collected over 4 $\sim$ 9 months with at least one week interval between the successive sessions. Therefore, the challenge from the long-term intra-speaker variation for speaker recognition can be addressed by the database.

4.2 Identification Results

Table 2 illustrates speaker identification results of the 4 systems use (i) WOCOR only, (ii) MFCC only, (iii) score fusion with fixed weight, and (iv) score fusion with confidence measure. The identification error rate (IDER) is defined as

$$IDER = \frac{Number\ of\ incorrrect\ identification\ trials}{Number\ of\ total\ identification\ trials} \times 100\% \qquad (10)$$

Table 2. Speaker identification performances

systems	MFCC	WOCOR	Fixed weight fusion	Fusion with CM
IDER (in%)	6.80	27.04	4.70	4.10

As illustrated, the identification performance of WOCOR is much worse than that of MFCC. Nevertheless, as expected, the fusion of these two complementary features gives better performance than that using MFCC only. For example, the IDER is reduced from 6.80% to 4.70% with fixed weight score fusion, and is further reduced to 4.10% with confidence measure based score fusion. As a whole, the proposed system provides an improvement of 40% over that using MFCC only.

4.3 Analysis of the Identification Results

Table 3 elaborates how the integration of the two complementary features affects the identification performances. The identification trials are divided into 4 subsets according to the performances of MFCC and WOCOR: (i) correct identification with both MFCC and WOCOR (McWc), (ii) incorrect identification with both MFCC and WOCOR (MiWi), (iii) incorrect identification with MFCC while correct identification with WOCOR (MiWc), and (iv) correct identification with MFCC while incorrect identification with WOCOR (McWi). The number of identification errors with MFCC, WOCOR and the integrated systems of each subset are given in the table.

Table 3. Number of errors of 4 identification subsets by different systems

Subsets	McWc	MiWi	MiWc	McWi
Numbers of trials	3328	244	95	1333
MFCC	0	244	95	0
WOCOR	0	244	0	1333
Fixed weight fusion	0	163	7	65
Fusion with CM	0	167	19	19

We are only interested in the last 3 subsets, which have errors with at least one kind of features. For the second subset, although both MFCC and WOCOR give incorrect identification results, the integrated system gives correct results for some trials. For example, the number of identification errors is reduced from 244 to 163 with the fixed weight fusion and to 167 with the confidence measure based fusion. That is, about one third of the errors has been corrected. Table 4 gives an example demonstrating how the score fusion can give correct result even though both MFCC and WOCOR give error results. In this example, the true speaker is S5. It is shown that although S5 only ranks at the 6th and the 2nd with MFCC and WOCOR respectively, in the integration systems, it ranks at the first and therefore is correct identified.

Table 4. Ranking the speaker scores in an identification trial

Rank	MFCC	WOCOR	Fixed weight fusion	Fusion with CM
1	S7: -1.7718	S34:1.5732	**S5:-0.4364**	**S5:-1.0903**
2	S27:-1.7718	**S5:1.5730**	S27:-0.4445	S27:-1.0977
3	S10:-1.7722	S48:1.5640	S34:-0.4446	S7: -1.0984
4	S42:-1.7743	S35:1.5620	S41:-0.4448	S10:-1.1000
5	S1: -1.7756	S39:1.5619	S46:-0.4448	S41:-1.1005
6	**S5:-1.7760**	S46:1.5510	S7: -0.4452	S46:-1.1015
7	S41:-1.7788	S41:1.5561	S10:-0.4465	S42:-1.1027

The results of the two one-error identification subsets McWi and MiWc demonstrate the superiority of the confidence measure for score fusion over the fixed weight score fusion. For the fixed weighting system, although the number of errors in the MiWc subset is significantly reduced from 95 to 7, there are 65 errors introduced to the McWi subset, which have been correctly identified with MFCC only. For the confidence measure based fusion system, the number of newly introduced errors is significantly reduced to 19, with only a slightly increase of errors in MiWi and MiWc subsets. As a whole, the total error number is reduced from 399 with MFCC only to 235 with fixed weight fusion, and further to 205 with confidence measure based fusion.

5 Conclusions

This paper presents an efficient information fusion technique to integrate two acoustic features MFCC and WOCOR for speaker identification. Analysis of the identification results shows that the two features have complementary contributions to speaker identification. To take full advantage of the complementary contributions of MFCC and WOCOR, a confidence measure derived from the speaker discrimination ratio of the two features is adopted as the weighting parameter for score level fusion of MFCC and WOCOR. In comparison with the identification error rate of 6.80% obtained with MFCC only, an error rate of 4.10% is obtained with the proposed information fusion system, that is an improvement of 40%.

Acknowledgements

Part of this work is supported by a RGC Grant #CUHK 4236/04E awarded by the Hong Kong Research Grant Council.

References

1. J. P. Campbell, "Speaker recognition: a tutorial," *Proc. IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.
2. D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

3. M. K. Sonmez, L. Heck, M. Weintraub, and E. Shriberg, "A lognormal tied mixture model of pitch for prosody based speaker recognition," in *Proc. Eurospeech*, 1997, pp. 1391–1394.
4. B. Imperl, Z. Kacic, and B. Horvat, "A study of harmonic features for speaker recognition," *Speech Communication*, vol. 22, no. 4, pp. 385–402, 1997.
5. M. D. Plumpe, T. F. Quatieri, and D. A. Reynolds, "Modeling of the glottal flow derivative waveform with application to speaker identification," *IEEE Trans. Speech Audio Processing*, vol. 7, no. 5, pp. 569–585, 1999.
6. Douglas Reynolds, Walter Andrews, Joseph Campbell, Jiri Navratil, Barbara Peskin, Andre Adami, Qin Jin, David Klusacek, Joy Abramson, Radu Mihaescu, Jack Godfrey, Doug Jones1, and Bing Xiang, "The SuperSID project: Exploiting high-level information for high-accuracy speaker recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, 2003, pp. 784–787.
7. N. H. Zheng, P. C. Ching, and Tan Lee, "Time frequency analysis of vocal source signal for speaker recognition," in *Proc. Int. Conf. on Spoken Language Processing*, 2004, pp. 2333–2336.
8. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, 1978.
9. S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.
10. A. Ross, A. Jain, and J.-Z. Qian, "Information fusion in biometrics," in *Proc. 3rd Int. Conf. Audio- and Video-Based Person Authentication (AVBPA)*, 2001, pp. 354–359.
11. D. Garcia-Romero, J. Fierrez-Aguilar, J. Gonzalez-Rodriguez, and J. Ortega-Garcia, "On the use of quality measures for text-independent speaker recognition," in *ESCA Workshop on Speaker and Language Recognition, Odyssey*, 2004, pp. 105–110.
12. Kar-Ann Toh and Wei-Yun Yau, "Fingerprint and speaker verification decisions fusion using a functional link network," *IEEE Trans. System, Man and Cybernetics B*, vol. 35, no. 3, pp. 2005, 357-370.
13. D. Talkin, "A robust algorithm for pitch tracking (RAPT)," *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal Eds. Elsevier, 1995.
14. I. Daubechies, *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, 1992.
15. N. H. Zheng, C. Qin, Tan Lee, and P. C. Ching, "CU2C: A dual-condition cantonese speech database for speaker recognition applications," in *Proc. Oriental-COCOSDA*, 2005, pp. 67–72.

Discriminative Transformation for Sufficient Adaptation in Text-Independent Speaker Verification

Hao Yang[2], Yuan Dong[1,2], Xianyu Zhao[1], Jian Zhao[2], and Haila Wang[1]

[1] France Telecom Research & Development Center, Beijing, 100083,
[2] Beijing University of Posts and Telecommunications, Beijing, 100876
{haoyang.hy, michaeljianzhao}@gmail.com,
{yuan.dong, xianyu.zhao, haila.wang}@francetelecom.com.cn

Abstract. In conventional Gaussian Mixture Model – Universal Background Model (GMM-UBM) text-independent speaker verification applications, the discriminability between speaker models and the universal background model (UBM) is crucial to system's performance. In this paper, we present a method based on heteroscedastic linear discriminant analysis (HLDA) that can enhance the discriminability between speaker models and UBM. This technique aims to discriminate the individual Gaussian distributions of the feature space. After the discriminative transformation, the overlapped parts of Gaussian distributions can be reduced. As a result, some Gaussian components of a target speaker model can be adapted more sufficiently during Maximum a Posteriori (MAP) adaptation, and these components will have more discriminative capability over the UBM. Results are presented on NIST 2004 Speaker Recognition data corpora where it is shown that this method provides significant performance improvements over the baseline system.

Keywords: Discriminative Adaptation HLDA EM ML.

1 Introduction

The Gaussian Mixture Model – Universal Background Model framework [1] is widely used in most state-of-the-art text-independent speaker verification systems. In a GMM-UBM system, the UBM is a single, speaker-independent GMM with a large number of mixture components (512-2048) trained from a large amount of speech uttered by lots of speakers. Due to the limitation of training speech data from an enrolled speaker, a target speaker model is often derived from adapting the parameters of UBM using the speaker's enrollment speech. Bayesian adaptation or MAP estimation is generally used for speaker adaptation [1]. In conventional MAP adaptation, a feature vector is used to update all Gaussian components in parallel. As the Gaussian distributions have overlapped parts, each feature vector is shared by several dominant components. With limited adaptation data, these components can not be adapted sufficiently, so the discriminative capability over UBM is reduced. This would impair subsequent verification performance.

To address this problem, one solution is to discriminate the distribution of the mixture components, which will reduce the overlapped sections of the Gaussian

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 558 – 565, 2006.

distributions. This idea is inspired by linear discriminative analysis, which is commonly used in pattern recognition. Linear discriminative analysis is to generate a "good" set of features from the observed data, and these features keep all class discriminating information. Features containing no information are removed since they will add computational and memory costs, and may degrade the classification results. In speech recognition, LDA [2] and HLDA [2,3] are often used.

In this paper, discriminative transformation for sufficient adaptation (DTFSA) is implemented in conventional GMM-UBM speaker verification system. This transformation is calculated using HLDA, but without dimension reduction, and it is used to discriminate the Gaussian components in the feature space. The method is tested on NIST SRE04 evaluation corpora, experimental results show that after the discriminative transformation, speaker models derived from MAP adaptation have more discriminative capability over the UBM, and the system gains a significant improvement on verification performance.

In the next section we will firstly review MAP adaptation and standard HLDA, and then describe using HLDA to get the discriminative transformation for sufficient adaptation in GMM-UBM system carefully. This is followed by a description of experiment data, design and results. Conclusion will be given at the end of this paper.

2 Discriminative Transformation for Sufficient Adaptation

2.1 MAP Adaptation

In the context of a GMM-UBM system for text-independent speaker recognition, consider a UBM with M components in which the model parameters λ are defined as $\lambda = \{w_m, \mu_m, \Sigma_m; m = 1, \cdots, M\}$, where w_m, μ_m and Σ_m are the component weight, mean vector and covariance matrix of the m-th component respectively. The adapted speaker model via MAP is obtained by updating the background model parameters for mixture components using new statistics collected from the adaptation data $X = \{x_1, \cdots, x_T\}$ [1], i.e., (in our following discussion and experiments, we focus on mean adaptation only)

$$\hat{\mu}_m = \alpha_m E_m(X) + (1 - \alpha_m)\mu_m, \tag{1}$$

where the new sufficient statistics are calculated as

$$\gamma_m = \sum_{t=1}^{T} \gamma_m(t) \tag{2}$$

$$E_m(X) = \sum_{t=1}^{T} \gamma_m(t) x_t \Big/ \gamma_m \,. \tag{3}$$

In the above equations, $\gamma_m(t)$ is the a posteriori probability of the m-th component given observation data x_t, which is

$$\gamma_m(t) = \frac{w_m N(x_t; \mu_m, \Sigma_m)}{\sum_j w_j N(x_t; \mu_j, \Sigma_j)}. \tag{4}$$

The adaptation coefficient α_m which controls the weight of a priori information is defined as follows

$$\alpha_m = \gamma_m / (\gamma_m + f), \tag{5}$$

where f is a fixed "relevance" factor (chosen to be 16 in all our experiments).

2.2 Standard HLDA Projection

Heteroscedastic linear discriminant analysis, which was first proposed by N.Kumar [2], is generally used in speech recognition. HLDA assumes n-dimensional original feature space can be split into two subspaces [2]: one subspace consists of p useful dimensions; the other subspace consists of another (n-p) nuisance dimensions. In the discriminative subspace, classes are well separated, while in the non-discriminative subspace, distributions of classes are overlapped, so they can not be separated completely. HLDA aims to find a projection matrix A, and use this matrix to map the original feature space to the new discriminative feature space. For a n-dimensional feature vector x, the transform can be written as

$$\bar{\mathbf{x}} = \mathbf{Ax} = \begin{bmatrix} \mathbf{A}_{[p]}\mathbf{x} \\ \mathbf{A}_{[n-p]}\mathbf{x} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{x}}_{[p]} \\ \bar{\mathbf{x}}_{[n-p]} \end{bmatrix} \tag{6}$$

The distribution of class j is modeled by a Gaussian distribution in rotated space, which is

$$p_j(\mathbf{x}) = \frac{\det(\mathbf{A})}{\sqrt{(2\pi)^n \det(\bar{\Sigma}^{(j)})}} \exp\left(-\frac{\left(\mathbf{Ax} - \bar{\mu}^{(j)}\right)^T \bar{\Sigma}^{(j)-1} \left(\mathbf{Ax} - \bar{\mu}^{(j)}\right)}{2} \right) \tag{7}$$

where $\bar{\mu}^{(j)}$ and $\bar{\sum}^{(j)}$ are mean vector and covariance matrix of class j in rotated space.

The maximum likelihood estimate of transformation A is given by Gales in [3]:

$$\mathbf{a}_k = \mathbf{c}_k \mathbf{G}^{(k)^{-1}} \sqrt{\frac{N}{\mathbf{c}_k \mathbf{G}^{(k)^{-1}} \mathbf{c}_k^T}} \tag{8}$$

where $\mathbf{c}_k$ is the k-th row of cofactor matrix of A. N is the total number of training vectors. And $\mathbf{G}^{(k)}$ is given by

$$\mathbf{G}^{(k)} = \begin{cases} \sum_{j=1}^{J} \frac{N_j}{\hat{\bar{\sigma}}_k^{(j)^2}} \Sigma^{(j)} & k \le p \\ \frac{N}{\hat{\bar{\sigma}}_k^{(j)^2}} \Sigma & k > p \end{cases} \tag{9}$$

Here, $\Sigma^{(j)}$ is covariance matrix of class j in original space. Σ is global covariance matrix of all the classes in original feature space. N_j is the number of training vectors belonging to class j . $\hat{\bar{\sigma}}_k^{(j)^2}$ is the k-th element of maximum likelihood estimate of the diagonal covariance matrix in the rotated space in terms of fixed transformation A .

2.3 Using HLDA to Get Discriminative Transformation for Sufficient Speaker Adaptation

From the derivation in section 2.1, it can be seen clearly that mixture components with large amounts of adaptation data would rely more on the new statistics and be well adapted to the target speaker.

But when using a GMM to model a speaker, Gaussian components of the model have cross sections, a given observation data x_t will be shared by several components. This will lead to the statistics collected for these components are not sufficient, and the discriminative capability of each component is deteriorated. If the overlapped parts of the Gaussian components are diminished, we can obviously find that the statistics collected for some components will be more sufficient, so they will enhance the speaker model's discriminative capability over the UBM. When an impostor test utterance is given, if the utterance has phonetic sounds corresponding to these well adapted components, the distance between distribution of the observation data and speaker model will increase, so the output impostor score will be smaller. This will improve system's performance.

As we mentioned before, heteroscedastic linear discriminant analysis (HLDA) [2, 3] is often used to discriminate different classes in feature space. In our method, HLDA without dimension reduction is used to find a series of new basis of the feature space that can minimize the overlapped sections of the Gaussian components.

In a GMM-UBM speaker verification system, a GMM with a large number of components is used to model a speaker's phonetic sound distribution, so we can assume each component stands for an individual acoustic class. Given transformation matrix A and model parameters, the distribution of the m-th component under the new basis can be written as:

$$p_m(\mathbf{x}) = \frac{\det(\mathbf{A})}{\sqrt{(2\pi)^D \det(\bar{\Sigma}^{(m)})}} \exp\left(-\frac{\left(\mathbf{A}\mathbf{x} - \bar{\mu}^{(m)}\right)^T \bar{\Sigma}^{(m)-1} \left(\mathbf{A}\mathbf{x} - \bar{\mu}^{(m)}\right)}{2} \right) \tag{10}$$

For computation efficiently, diagonal covariance is often used. The optimal solution of transformation A can be achieved using ML framework [3]. But unlike Gales, we do not split the feature space into useful sub-space and nuisance sub-space. The EM auxiliary function is given by

$$Q(\lambda,\hat{\lambda})=\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_m(t)\log\left\{\frac{\det(\mathbf{A})}{\sqrt{(2\pi)^D\det(\overline{\Sigma}_{diag}^{(m)})}}\exp\left(-\frac{\left(\mathbf{A}x_t-\overline{\mu}^{(m)}\right)^T\overline{\Sigma}_{diag}^{(m)^{-1}}\left(\mathbf{A}x_t-\overline{\mu}^{(m)}\right)}{2}\right)\right\} \tag{11}$$

where $\lambda=\left\{\overline{\mu}^{(m)},\overline{\Sigma}_{diag}^{(m)},A\right\}$ is the "old" parameter, and $\hat{\lambda}=\left\{\hat{\overline{\mu}}^{(m)},\hat{\overline{\Sigma}}_{diag}^{(m)},\hat{A}\right\}$ is the "new" parameter optimized respected to the old parameter. Eq(11) has no close-form solution, but Gales' scheme [3] can be used. Firstly, we can find the maximum likelihood estimate of $\overline{\mu}^{(m)}$ and $\overline{\Sigma}_{diag}^{(m)}$ in terms of fixed transformation A by differentiating eq(11):

$$\hat{\overline{\mu}}^{(m)}=A\overline{\mu}^{(m)}A^T \tag{12}$$

$$\hat{\overline{\Sigma}}_{diag}^{(m)}=diag\left(A\Sigma^{(m)}A^T\right)=\left\{\hat{\overline{\delta}}_d^{(m)^2}\right\} \tag{13}$$

where $\Sigma^{(m)}$ is the covariance matrix of m-th Gaussian component in original feature space.

Substitute the maximum estimation of $\overline{\mu}^{(m)}$ and $\overline{\Sigma}_{diag}^{(m)}$ into eq(11), and eq(11) can be rewritten as

$$Q(A,\hat{A})=\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_m(t)\left\{\log\frac{\det(\mathbf{A})}{\sqrt{(2\pi)^D\det(\hat{\overline{\Sigma}}_{diag}^{(m)})}}-\sum_{d=1}^{D}\frac{a_d\Sigma^{(m)}a_d^T}{2\hat{\overline{\delta}}_d^{(m)^2}}\right\} \tag{14}$$

In eq(14), a_d is the d-th row of the transformation A.

By differentiating eq(14) with respect to a_d in terms of the fixed current estimates of variances $\hat{\overline{\Sigma}}_{diag}^{(m)}$, we can find the maximum likelihood estimate of a_d:

$$\hat{a}_d=c_dG_d^{-1}\sqrt{\frac{T}{c_dG_d^{-1}c_d^T}} \tag{15}$$

where

$$G_d=\sum_{m=1}^{M}\frac{\gamma_m\Sigma^{(m)}}{\hat{\overline{\delta}}_d^{(m)^2}}. \tag{16}$$

and T is the total number of training vectors. γ_m is calculated by eq(2).

After the transformation, the overlapped sections of the Gaussian distributions in the feature space are reduced, this will make the speaker adaptation more sufficient for some components. As we show in section 3, this method will improve system's performance.

3 Experimental Results

In this section, we report speaker verification experiments conducted on 1-side training, 1-side testing (1conv4w-1conv4w) part of NIST 2004 speaker recognition evaluation dataset, which is a multi-language data corpus. The language of this data corpus contains Mandarin, English, Arabic, etc. This evaluation condition includes 616 speakers (248 male and 368 female). For each speaker, approximately 2.5 minutes of speech from a single telephone call is used for enrollment. Verification utterances are also about 2.5 minutes of speech from only one telephony conversation. There are totally 1174 verification utterances. Each verification utterance is scored against a number of designated putative speaker models. There are totally 26224 verification trials.

For our baseline system, 14-dimensional MFCC vectors are extracted from the silence removed speech signal every 10ms using 25ms window. Bandlimiting is performed by only retaining the filterbank outputs from the frequency range 300Hz-3400Hz. Cepstral features are processed with RASTA [4] filtering and Feature Mapping [5] to eliminate channel distortion. Delta, acceleration and triple-delta coefficients are then computed over ± 2 frames span and appended to the static coefficients, producing a 56 dimensional feature vector.

The background model used for all targets is a gender independent 2048 mixture GMM trained using data from Switchboard II database. Target models are derived by Bayesian adaptation (a.k.a. MAP estimation) of the UBM parameters using the designated training data. Based on observed better performance, only the mean vectors are adapted. The relevance factor is set to 16. For Feature Mapping, gender and channel-dependent models are also adapted from the UBM.

Results are presented using Histogram and Detection Error Tradeoff (DET) plots. Performance is computed after collecting all verification scores. Along with Equal Error Rate (EER), the minimum decision cost function (DCF), defined by NIST as DCF = 0.1 * Pr(miss) + 0.99 * Pr(false_alarm) [6], is also used as an overall performance measure.

Fig.1 gives the histogram of target and impostor score for baseline system and DTFSA system. From this figure, it can be seen that the impostor scores of DTFSA system are obvious smaller than the baseline system, so the discriminability between target speakers and impostor speakers (modeled by UBM) is enhanced.

In Fig.2, we show DET curves for baseline system and DTFSA system. We can see DTFSA produces significant gains over the baseline system. The EER of "DTFSA" reduces from 13.7% to 12.9%, while the minimum DCF value drops from 59.2×10^{-3} to 53.7×10^{-3}.

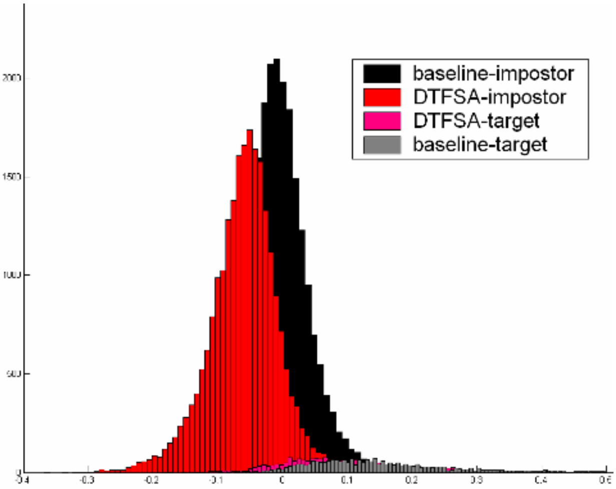


Fig. 1. Histograms of target and impostor scores for baseline system and DTFSA system

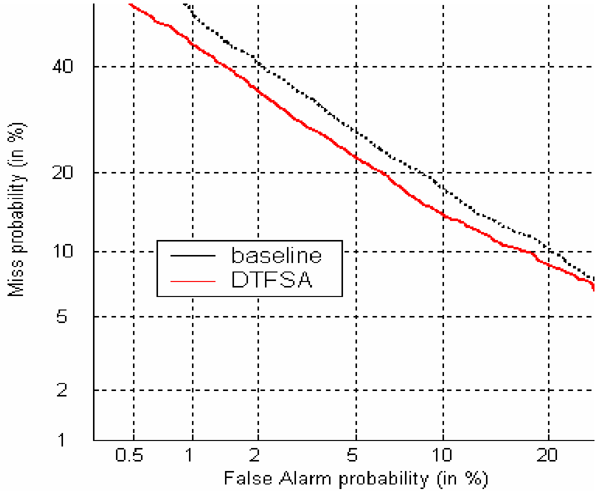


Fig. 2. DET curves for baseline system and DTFSA system

4 Conclusion

A discriminative transformation for sufficient speaker adaptation based on HLDA is implemented in this paper. Standard MAP adaptation can not model speaker's character well with limited training data due to the overlapped sections of individual Gaussian components. With the discriminative transformation, the overlapped sections of Gaussian components are reduced, so some components of target speaker model can be adapted more sufficiently, this will increase speaker model's discriminative capability over UBM.

The experimental results using the NIST speaker recognition evaluation dataset show that better verification performance is obtained with models adapted through this new method. Further experiments with more extensive datasets are planned in future work.

References

1. D.A.Reynolds, T.Quatieri, R.Dunn.: Speaker Verification Using Adapted Mixture Models. Digital Signal Processing, Vol.10. (2000) 181-202
2. N. Kumar.: Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition. Ph.d. thesis, John Hopkins University, Baltimore, USA (1997)
3. M.J.F. Gales.: Maximum likelihood multiple projection schemes for hidden Markov models. Technical Report CUED/F-INFENG/TR.365, Cambridge University, UK (1999)
4. H. Hermansky, N. Morgan.: RASTA Processing of Speech. IEEE Trans. on Speech and Audio Processing, vol.2. (1994) 578-589
5. D.A.Reynolds.: Channel robust speaker verification via feature mapping. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol.2. (2003) 53-56
6. The NIST 2004 Speaker Recognition Evaluation Plan. [Online]. Available: http://www.nist.gov/speech/tests/spk/

Fusion of Acoustic and Tokenization Features for Speaker Recognition

Rong Tong[1,2], Bin Ma[1], Kong-Aik Lee[1], Changhuai You[1], Donglai Zhu[1],
Tomi Kinnunen[1], Hanwu Sun[1], Minghui Dong[1], Eng-Siong Chng[2], and Haizhou Li[1,2]

[1] Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613
tongrong@i2r.a-star.edu.sg
[2] School of Computer Engineering,
Nanyang Technological University, Singapore 639798
asechng@ntu.edu.sg

Abstract. This paper describes our recent efforts in exploring effective discriminative features for speaker recognition. Recent researches have indicated that the appropriate fusion of features is critical to improve the performance of speaker recognition system. In this paper we describe our approaches for the NIST 2006 Speaker Recognition Evaluation. Our system integrated the cepstral GMM modeling, cepstral SVM modeling and tokenization at both phone level and frame level. The experimental results on both NIST 2005 SRE corpus and NIST 2006 SRE corpus are presented. The fused system achieved 8.14% equal error rate on 1conv4w-1conv4w test condition of the NIST 2006 SRE.

Keywords: speaker recognition, cepstral feature, phonotactic feature, Gaussian mixture model, support vector machine, tokenization, fusion.

1 Introduction

In the past decade, much progress has been made in text-independent speaker recognition using acoustic features such as Gaussian Mixture Modeling (GMM) on amplitude spectrum based features [1] and Support Vector Machine (SVM) on Shifted Delta Cepstral (SDC) [2]. In recent years, some tokenization methods with higher level information have attracted great interests. These tokenization methods convert the speech into different levels of tokens, such as words, phones and GMM tokens. For example, lexical features based on word n-grams has been studied in [3] for speaker recognition; Parallel Phone Recognition followed by Language Modeling (PPRLM) [4] has been extensively adopted in language and speaker recognition; Gaussian Mixture Model Tokenization [5], [6] has been used with the tokens at the frame level for language and speaker recognition.

Related works generally agree that the integration of features with different degrees of discriminative information can improve the performance of speaker recognition system and that the appropriate fusion technique is critical.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 566 – 577, 2006.

The acoustic features, such as MFCC features, adopted in speech recognition systems have also used in speaker recognition. The Gaussian mixture modeling (GMM) based on MFCCs has demonstrated a great success for text-independent speaker recognition [1]. To model the out-of-set speakers, a universal background model (UBM) is used to normalize the likelihood scores from different speakers. The model of a specific speaker is obtained with Bayesian adaptation based on UBM by using the training data of that speaker [1]. Test normalization (Tnorm) [7] is the technique to align the score distributions of individual speaker models. The score mean and variance of multiple non-target speaker models are used to normalize the score of the target speaker models. To capture the long time dynamic information, we used temporal discrete cosine transform (TDCT) feature [8] in the GMM framework.

Support vector machine (SVMs) is widely used in many pattern classification tasks. A SVM is a discriminative classifier to separate two classes with a hyperplane in a high-dimensional space. In [2], the generalized linear discrininant sequence kernel (GLDS) for the SVM is proposed for speaker and language recognition. The feature vectors extracted from an utterance are expanded to a high-dimensional space by calculating all the monomials. Past works [2] also show that the front-end with linear prediction cepstral coefficients (LPCCs) gives better performance than the front-end with MFCCs. We construct two SVM subsystems based on both MFCCs and LPCCs.

Recently, researches using phonotactic features showed that it can provide effective complementary cues for speaker and language recognition. The phonotactic features are extracted from an utterance in the form of tokens. The tokens may be at different levels, words, phones and even frames. PPRLM [4] uses multiple parallel phone recognizers to convert the input utterance into phone token sequence and the sequence is processed by a set of *n*-gram phone language models. In this paper, instead of using *n*-gram phone language models, we propose to use vector space modeling (VSM) as the backend classifier [9]. For each phone sequence generated from the multiple parallel phone recognizers, we count the occurrences of phone *n*-grams. Thus, a phone sequence is then represented as a high-dimensional vector of *n*-gram occurrences known as Bag-of-Sounds (BOS) vectors. The SVM is used as the classifier.

The tokenization can also be made at the frame level, such as Gaussian Mixture Model Tokenization [5] for language identification. Tokenization at the frame level captures another aspect of acoustic and phonetic characteristics among the languages and speakers. It also provides more tokens than the phone recognizers from the limited speech data. Similar to PPRLM, we use multiple parallel GMM tokenizers to improve speaker coverage in speaker recognition. We propose to use speaker cluster based GMM tokenization as one of the subsystems in our speaker recognition system that multiple GMM tokenizers are constructed according to the speaker characteristics.

This paper is organized as follows. Section 2 introduces the speech corpora used. Section 3 describes our six subsystems and the score fusion strategy. Section 4 presents the experimental results on the development data (NIST 2005 SRE) as well as on the NIST 2006 SRE data and section 5 presents our conclusions.

2 Our Submission and Speech Corpora

The NIST 2006 SRE evaluation task is divided into 15 distinct and separate tests. Each of these tests involves one of the five training conditions and one of four test conditions [10]. The five training conditions are 10-second speech excerpt from a two-channel/4-wire (10sec4w), one conversation side of approximately five minutes total duration from a two-channel/4-wire (1conv4w), three conversation sides (3conv4w), eight conversation sides (8conv4w) and three conversation sides from a summed-channel/2-wire (3conv2w). The four test conditions are 10-second speech excerpt from two-channel/4 wire (10sec4w), one conversation side from a two-channel/4-wire (1conv4w), one conversation side from a summed-channel/2-wire (1conv2w) and 1 conversation side recorded by auxiliary microphone (1convMic).

The performance of the NIST speaker recognition system is evaluated by the detection cost function. It is defined as a weighted sum of miss and false alarm error probabilities:

$$C_{Det} = C_{Miss} \times P_{Miss|T\arg et} \times P_{T\arg et} + C_{FalseAlarm} \times P_{FalseAlarm|NonT\arg et} \times (1 - P_{T\arg et}) \qquad (1)$$

In NIST 2006 SRE, $C_{Miss} = 10$, $C_{FalseAlarm} = 1$ and $P_{T\arg et} = 0.01$. The experiment results presented in this paper are reported in Equal Error Rate (EER) and DET curves. EER is used to decide the operating point when the false acceptance rate (FAR) and false rejection rate (FRR) are equal.

2.1 Our Submission for the NIST 2006 SRE

Our speaker recognition system, IIR (Institute for Infocomm Research) speaker recognition system, participated in seven tests out of the 15 evaluation tasks. The tasks we participated involve 4 training conditions and 2 test conditions: 10sec4w-10sec4w, 1conv4w-10sec4w, 3conv4w-10sec4w, 8conv4w-10sec4w, 1conv4w-1conv4w, 3conv4w-1conv4w and 8conv4w-1conv4w. Table 1 shows the available tasks in the NIST 2006 SRE and the 7 tests that we participated.

There are six subsystems in our speaker recognition system for the NIST 2006 SRE. These subsystems fall into three categories: (i) spectral features with SVM modeling including MFCC feature based spectral SVM (Spectral MFCC-SVM) and LPCC feature based spectral SVM (Spectral LPCC-SVM); (ii) spectral feature with GMM modeling including MFCC feature based GMM (MFCC-GMM) and TDCT feature [8] based GMM (TDCT-GMM); (iii) tokenization features with vector space modeling (VSM) including parallel phone recognizers based tokenization: Bag-of-Sounds (BOS) and speaker clustering based multiple GMM tokenizers (GMM token). The first four subsystems capture the characteristics of spectral features while the last two tokenization subsystems capture the phonotactic information. Fig. 1 shows the system framework of our submission. The six subsystems scores are fused to make the final decision.

Table 1. The seven tests that we participated in the NIST 2006 SRE

		Test segment condition			
		10sec 2-chan	**1conv 2-chan**	**1conv summed chan**	**1 conv aux mic**
Training condition	**10sec 2-chan**	10sec4w-10sec4w			
	1conv 2-chan	1conv4w-10sec4w	1conv4w-1conv4w	N.A.	N.A.
	3conv 2-chan	3conv4w-10sec4w	3conv4w-1conv4w	N.A.	N.A.
	8conv 2-chan	8conv4w-10sec4w	8conv4w-1conv4w	N.A.	N.A.
	3conv summed-chan		N.A.	N.A.	

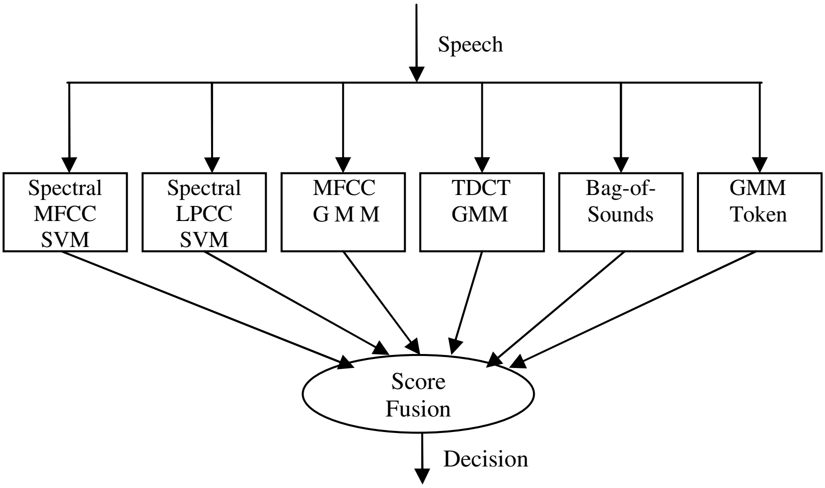


Fig. 1. System framework of our submission for the NIST 2006 SRE

2.2 Speech Corpora

Table 2 shows the training and development data for each subsystem. The tokenizer training data are used to model the parallel phone recognizers or to model the parallel GMM tokeniers. Background speaker data are used to train UBM models or to train speaker background models. Cohort data are used to make the Test normalization (Tnorm). The NIST 2005 SRE corpus is used to evaluate the performance of individual systems. The output scores of the six subsystems are used to train the score fusion to facilitate the final decisions on the NIST 2006 SRE data. We will describe each of these speech corpora in the next section together with the subsystems.

Table 2. Speech corpora for the training and development

	LPCC-SVM	MFCC-SVM	MFCC-GMM	TDCT-GMM	BOS	GMM-Token
Tokenizer training data	N.A.				1. IIR-LID corpus 2. LDC Korean corpus 3. MAT corpus 4. OGI-TS corpus	NIST 2002 SRE corpus
Background speaker data (UBM)	Switchboard corpora: sw3p1, sw3p2, sw2p2 and sw2p3		NIST2004 1side training files	NIST2004 1side training files	NIST 2002 SRE corpus	NIST 2004 SRE corpus
Cohort data (Tnorm)	Evaluation set of the NIST 2004 SRE corpus		N.A.			
Development /Test	NIST 2005 SRE corpus					

3 System Description

For the spectral SVM and GMM subsystems, an energy-based voice activity detector (VAD) is applied after feature extraction to remove non-speech frames. We train two GMMs with 64 mixture components to model the energy distributions of the speech frames as well as the non-speech frames by using the development set of the NIST 2001 SRE corpus. A predefined energy threshold is used to make speech/non-speech frames separation. With such a VAD algorithm, about 38% speech and 62% non-speech frames were detected in the NIST 2006 SRE corpus.

For the Bag-of-Sounds and GMM token subsystems, the VAD algorithm chunks the long conversations into smaller utterances so that the tokenization methods can be applied to create phone or GMM token sequence for each of the utterances. The utterance based cepstral mean subtraction is performed to filter off the channel distortion.

3.1 Spectral LPCC-SVM and Spectral MFCC-SVM Subsystems

The support vector machine (SVM) is a two-class classifier, and for the speaker recognition task, it can be used to model the boundary between a speaker and a set of background speakers. The background speakers represent the population of imposters expected during recognition. We follow the work reported in [2] and [11] in which the generalized linear discriminant sequence kernel (GLDS) is used for speaker and language recognition.

Two kinds of acoustic spectral features, the MFCC features and LPCC features, both with a dimension of 36, are used in the two SVM subsystems. For the MFCC front-end, we use a 27-channel filterbank, and 12MFCC + 12Δ + 12$\Delta\Delta$ coefficients. For the LPCC front-end, 18LPCC + 18Δ coefficients are used.

The feature vectors extracted from an utterance is expanded to a higher dimensional space by calculating all the monomials up to order 3, resulting in a feature space expansion from 36 to 9139 in dimension. The expanded features are then averaged to form an average expanded feature vector for each of the utterances under consideration. In the implementation, it is also assumed that the kernel inner product matrix is diagonal for computational simplicity.

During enrollment, the current speaker under training is labeled as class +1, whereas a value of -1 is used for the background speakers. The set of background speaker data is selected from Switchboard 3 Phase 1 and Phase 2 (for Cellular data) and Switchboard 2 Phase 2 and Phase 3 (for landline telephone). We randomly select 2000 utterances from each of the 4 datasets to form the background speaker database of 8000 utterances, with roughly equal amounts of male and female speakers. Each utterance in the background and the utterance of the current speaker under training is represented with an average expanded feature vector b_{av}. These average expanded features are used in the SVM training. The commonly available toolkit SVMTorch [13] is used for this purpose. The result of the training is a vector w of dimension 9139 which represents the desired target speaker model [11]. During evaluation, an average expanded feature vector b_{av} is formed for each of the input utterances, and the score is taken as the inner product between these two vectors, i.e, $w^T b_{av}$.

Test normalization (Tnorm) method [7] is adopted in the two subsystems. The NIST 2004 training data is used to form the cohort models. In particular, the speaker models in the NIST 2004 are used as the cohort models. The training condition of the cohort models and evaluation corpus are matched. For example, the trained models in the 1side of NIST 2004 are used as the cohort models for the target models in the 1conv4w training condition of the NIST 2005 and 2006 SRE corpus. Similar concept is applied to 10sec4w, 3conv4w, and 8conv4w training conditions.

3.2 MFCC-GMM and TDCT-GMM Subsystems

Two kinds of spectral features are used in the two GMM modeling subsystems. One is the MFCC features same as those adopted for spectral SVM subsystem. Another one use the temporal discrete cosine transform (TDCT) features [8].

The conventional MFCC features characterize the spectral character in a short-time frame of speech (typically 20~30 ms). Psychoacoustic studies [13] suggest that the peripheral auditory system in humans integrates information from much larger time spans than the temporal duration of the frame used in speech analysis. Inspired by this finding, the TDCT feature is aiming at capturing the long time dynamic of the spectral features [8].

The MFCC feature vector in the MFCC-GMM system has 36 components comprising of 12 MFCC coefficients along with their first and second order derivatives. The TDCT feature vector has 108 in dimension. For both of the two GMM subsystems, the gender-dependent background models with 256 Gaussian mixtures are trained using the NIST2004 1-side training data subset. The target speaker models are adapted from these two background models. The background model having the same gender with the target speaker is used for adaptation. In the evaluation, the likelihood ratio between the target speaker model and the background model of the same gender is used as output score for the speaker.

3.3 Bag-of-Sounds Subsystem

This system uses the front end that consists of parallel phone tokenizers, and vector space modeling as the back end classifier [9].

Seven phone tokenizers of the following languages including English, Korean, Mandarin, Japanese, Hindi, Spanish and German. The English phone recognizer is trained from IIR-LID database [14]. The Korean phone recognizer is trained from LDC Korean corpus (LDC2003S03). The Mandarin phone recognizer is trained from MAT corpus [15]. And the other four phone recognizers are trained from OGI-TS corpus [16]. Each phone is modeled with a three-state HMM, and 39-dimensional MFCC features are used. Each HMM state of the English, Korean and Mandarin languages are modeled with 32 Gaussian mixtures, while the states in other languages are with 6 Gaussian mixtures due to the availability of training data. Phone recognition is performed with the Viterbi search using a fully connected null-grammar network of phones.

For a given speech utterance, the tokenizers yield seven phone sequences. They are converted to a vector of weighted terms in three steps. Firstly, we compute unigram and bigram probabilities for each phone sequence, and then organize the probabilities into a vector. Secondly, each entry in the vector is multiplied by a background component [17], and finally, the vectors from each of the seven languages are concatenated to form the feature vector.

In the SVM training process, a single vector of weighted probabilities is derived from each conversation side. We use a one-versus-all strategy to train the SVM model for a given speaker. The conversation side of the target speaker is labeled as class +1, while all the conversation sides in the background are labeled as class -1. The NIST 2002 SRE corpus is used as background data. During the evaluation, the input utterance is converted to the feature vector and a score is produced from the SVM model. The toolkit SVMTorch [12] with a linear kernel is used.

3.4 GMM Token Subsystem

This system uses multiple GMM tokenizers as the front end, and vector space modeling as the back end classifier [6]. Each GMM tokenizer converts the input speech into a sequence of GMM token symbols which are indexes of the Gaussian components that score the highest for each frame in the GMM computation. The GMM token sequences are then processed in the same way as the process of phone sequences in the bag-of-sounds approach, i.e., the sequences are converted to a vector of weighted terms and then recognized by a speaker's SVM model.

Inspired by the finding of PPRLM in language recognition where multiple parallel single-language phone recognizers in the front-end enhance the language coverage and improve the language recognition accuracy over single phone recognizer, we explore multiple GMM tokenizers to improve speaker characteristics coverage and to provide more discriminative information for speaker recognition [6]. By clustering all the speakers in the training set into several speaker clusters, we represent the training space in several partitions. Each partition of speech data can then be used to train a GMM tokenizer. With each of these parallel GMM tokenizers, a speech segment is converted to a feature vector of weighted terms. The multiple feature vectors are then concatenated to form a composite vector for SVM modeling. The NIST 2002 SRE corpus is used for the training of speaker cluster based GMM tokenizers, and the NIST 2004 SRE corpus is used as the background data. 10 parallel GMM tokenizers, each having 128 mixtures of Gaussian components, are constructed.

3.5 Score fusion of Subsystems

The score of the six subsystems described above are combined using SVM classifiers as shown in Fig. 2. For a given speech utterance and the reference speaker, a 6-dimensional score vector is derived from the six subsystems. The score vectors are first normalized to zero mean and unit variance. Then the polynomial expansion of order 1, 2 and 3 are applied to the normalized score vectors. Three sets of expanded score vectors with dimension 7, 28 and 84 are obtained. Each set of the expanded score vectors are used to train a SVM model. The final decision is made according to the averaged value of three SVM scores.

The NIST 2005 SRE evaluation corpus is used as the training data for these three SVMs. The score vectors generated from the genuine utterances are labeled as class +1, and the score vectors generated from the impostor utterances are labeled as class -1. The thresholds estimated from the NIST 2005 SRE corpus are used for final True/False decision on the NIST 2006 SRE. The toolkit SVMTorch [12] with a radial kernel is used.

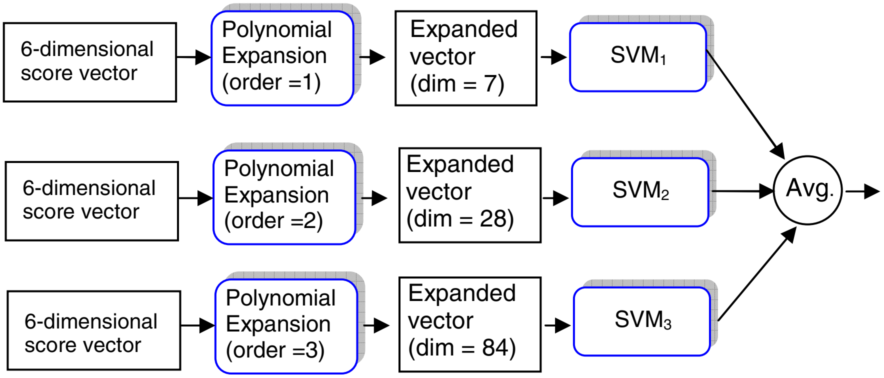

6-dimensional score vector
Polynomial Expansion (order =1)
Expanded vector (dim = 7)
SVM1
6-dimensional score vector
Polynomial Expansion (order =2)
Expanded vector (dim = 28)
SVM2
Avg.
6-dimensional score vector
Polynomial Expansion (order =3)
Expanded vector (dim = 84)
SVM3

Fig. 2. Score fusion of the six subsystems

4 Experiment Results

The NIST 2005 SRE evaluation set is used to evaluate the performance of the six subsystems before the system is finalized for the NIST 2006 SRE competition. It is also used as the development set to estimate the thresholds of the score fusion which provides the genuine/impostor decision for all the trials in the NIST 2006 SRE. Table 3 shows the equal error rates (EER %) of the six subsystems as well as the score fusion on seven test conditions in the NIST 2005 SRE.

Table 3. EER% of subsystems and fusion on the NIST 2005 SRE evaluation set (for the test conditions, the first part refers to training, the second part refers to testing, e.g, 1conv4w-10sec4w means training with 1conv4w, testing with 10sec4w)

Test / System	LPCC-SVM	MFCC-SVM	MFCC-GMM	TDCT-GMM	BOS	GMM-Token	Fusion
10sec4w-10sec4w	29.41	31.28	28.72	30.39	41.80	40.35	24.62
1conv4w-10sec4w	18.74	19.92	19.78	18.76	28.96	31.05	13.80
1conv4w-1conv4w	10.55	11.32	13.55	13.81	19.31	22.38	7.82
3conv4w-10sec4w	14.40	16.02	16.16	15.60	24.93	25.26	11.32
3conv4w-1conv4w	6.87	8.07	10.26	9.97	14.32	16.11	5.67
8conv4w-10sec4w	13.05	14.00	14.54	14.45	22.29	24.34	9.76
8conv4w-1conv4w	5.73	7.17	9.42	9.11	12.22	17.27	4.56

The results show that the four acoustic feature based subsystems outperform the two tokenization subsystems and the best subsystem is the LPCC-SVM system. Compared with the MFCC-GMM system, the TDCT-GMM system captures the longer time dynamic of the spectral features and it requires more training data. This explains why TDCT-GMM system achieves better accuracy than MFCC-GMM system when the amount of the training data increases. By combining the score of the six subsystems, the overall result improves significantly.

With the six subsystems and the thresholds of the score fusion obtained from the NIST 2005 SRE corpus, the NIST 2006 SRE data are processed. Fig. 3 shows the performance of the seven test conditions of the NIST 2006 SRE. The DET curves and the EER% for all the seven test conditions are illustrated. In the DET curves, the points of Min C-det denote the best results we can achieve from all possible thresholds for the final decision. The points of the actual decision denote the results on our actual designed thresholds.

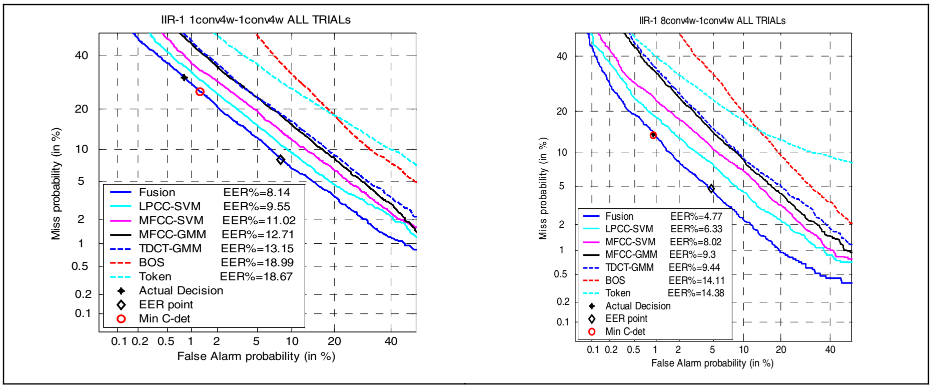


Fig. 3. Performance on the NIST 2006 SRE

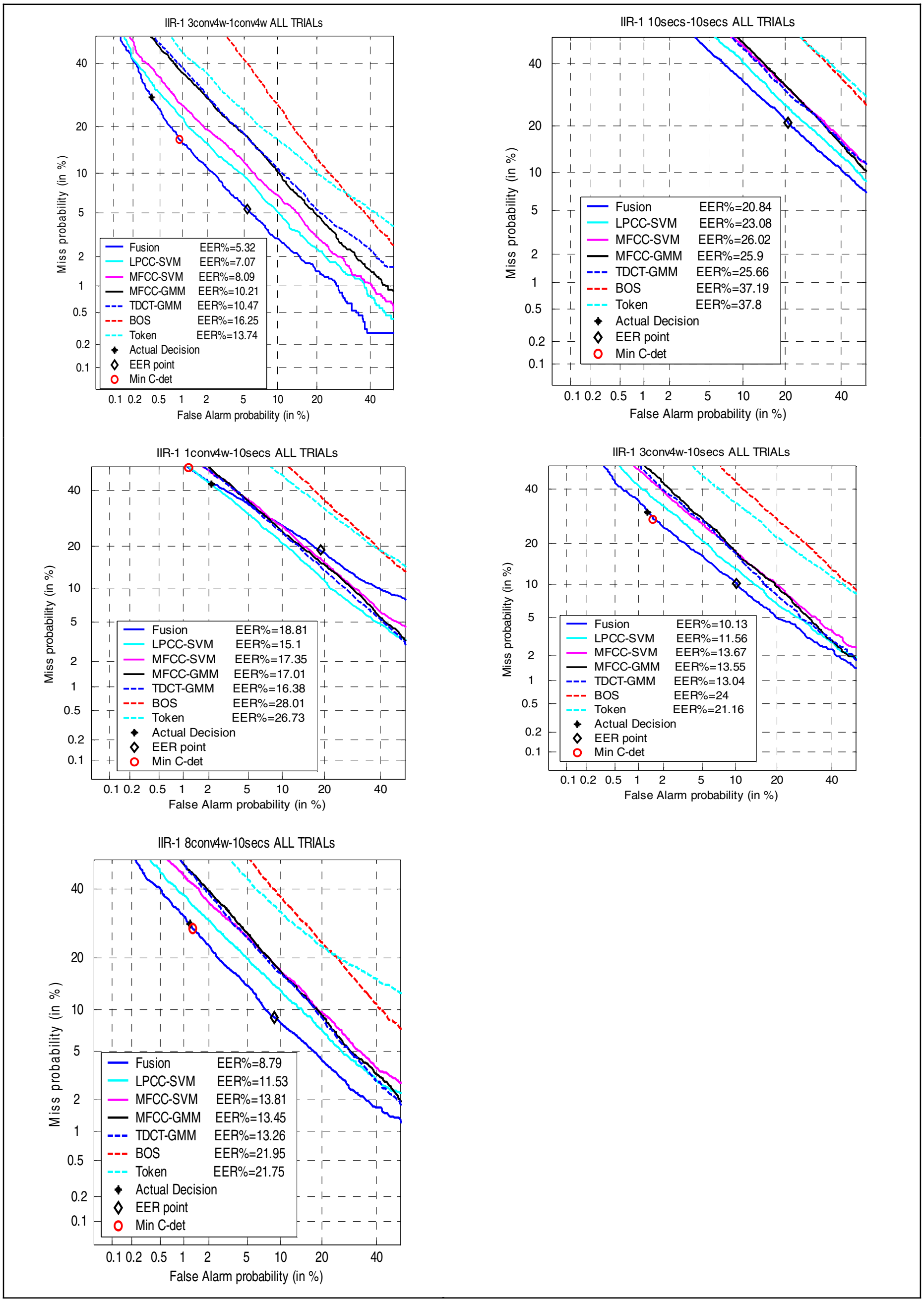


Fig. 3. (*continued*)

To compare the contribution of each subsystem category to the final fusion, we use the best subsystem, the spectral LPCC-SVM as the baseline. Three other subsystems, MFCC-SVM, MFCC-GMM and Bag-of-Sounds (BOS), will be combined with LPCC-SVM subsystem individually to examine the performance of the fusion. The experiments are conducted on three test conditions of NIST 2006 SRE, 10sec4w-10sec4w, 1conv-1conv4w and 8conv4w-1conv4w that involve three training segment conditions and two test segment conditions. Table 4 shows the results. The numbers in the bracket are relative EER reduction compared with the baseline system.

Since more information is provided, the combinations generally give us better performance. For the short test segment (10sec4w-10sec4w), the MFCC-GMM subsystem contributed the best error reduction. Although both MFCC-GMM and LPCC-SVM use acoustic features, they model the spectral features with different method and can make good use of more discriminative information. Bag-of-Sounds subsystem uses phonotactic features that provide complementary information to acoustic features for the speaker recognition task. A relative EER reduction of 22.6% has been achieved based on the LPCC-SVM subsystem on the 8conv4w-1conv4w test. The combination of LPCC and MFCC features with SVM method also produce better results in all the three test conditions.

Table 4. Performances (EER%) of different subsystem combinations on the NIST 2006 SRE, (for the test conditions, the first part refers to training, the second part refers to testing, e.g, 1conv4w-10sec4w means training with 1conv4w, testing with 10sec4w)

Test/ System	**LPCC-SVM**	**LPCC-SVM MFCC-SVM**	**LPCC-SVM MFCC-GMM**	**LPCC-SVM BOS**
10sec4w-10sec4w	23.08	22.94 (0.6%)	21.27 (7.8%)	23.47 (-1.7%)
1conv4w-1conv4w	9.55	8.72 (8.7%)	8.44 (11.6%)	8.78 (8.1%)
8conv4w-1conv4w	6.33	5.18 (18.2%)	5.07 (19.9%)	4.90 (22.6%)

5 Summary and Discussion

We present our speaker recognition system for NIST 2006 SRE. The system consists of six subsystems that capture both acoustic features and phonotactic information. For the acoustic features, both GMM modeling and spectral SVM modeling are adopted. Besides the conventional features, such as MFCCs and LPCCs, we propose to use TDCT features to model the long time dynamic of the spectral information. To capture speaker discriminative information from the higher level, tokenization methods are used to create phone token sequence and GMM token sequence from each of the utterances. For a given utterance, all the n-gram probabilities of the token sequence are calculated and combined into an n-gram statistic vector. A high dimensional vector is obtained by concatenating multiple token sequences generated from parallel phone recognizers or parallel GMM tokeniziers. Vector space modeling method is adopted as the backend classifier to model these high dimensional n-gram statistic vectors.

The experimental results show that the acoustic features are more effective in speaker recognition. The phonotactic features also provide complementary information and can improve the system performance significantly on longer speech segments. The experiment results on the subsystem fusion showed that the appropriate combination of the discriminative features from multiple sources is an effective method to improve the speaker recognition accuracy.

References

1. Reynolds, D. A., Quatieri, T. F. and Dunn, R. B.: Speaker Verification Using Adapted Gaussian Mixture Modeling. Digital Signal Processing, 10 (2000), pp. 19-41.
2. Campbell, W. M., Campbell, J. P., Reynolds, D. A., Singer, E. and Torres-Carrasquillo, P. A.: Support Vector Machines for Speaker and Language Recognition. Computer Speech and Language, 20 (2006), pp. 210-229.
3. Doddington, G.: Speaker Recognition based on Idiolectal Differences between Speakers. Proc. Eurospeech, 2001.
4. Zissman, M. A.: Comparison of Four Approaches to Automatic Language Identification of Telephone Speech. IEEE Trans. on Speech and Audio Processing, vol. 4, no. 1, 1996.
5. Torres-Carrasquillo, P. A., Reynolds, D. A. and Deller, Jr., J. R.: Language Identification using Gaussian Mixture Model Tokenization. Proc. ICASSP, 2002.
6. Ma, B., Zhu, D., Tong, R. and Li, H.: Speaker cluster based GMM tokenization for speaker recognition. To appear in Interspeech 2006.
7. Auckenthaler, R., Carey, M. and Lloyd-Thomas, H.: Score normalization for text-independent speaker verification systems. Digital Signal Processing, vol. 10, no 1-3, pp. 42-54, Jan 2000.
8. Kinnunen, T. H., Koh, C. W. E., Wang, L., Li, H. and Chng, E.S.: Temporal Discrete Cosine Transform: Towards Longer Term Temporal Features for Speaker Verification, accepted for presentation in 5th International Symposium on Chinese Spoken Language Processing, 2006.
9. Li, H. and Ma, B.: A Phonotactic Language Model for Spoken Language Identification", 43rd Annual Meeting of the Association for Computational Linguistics (ACL05), June 2005, Ann Arbor, USA.
10. http://www.nist.gov/speech/tests/spk/2006/sre-06_evalplan-v9.pdf
11. Campbell, W.M.: “Generalized linear discrininant sequence kernels for speaker recognition,” in Proc. ICASSP, pp. 161-164, 2002
12. Collobert, R. and Bengio, S.: SVMTorch: support vector machines for large-scale regression problems. Journal of Machine Learning Research, vol. 1, pp. 143-160, 2001.
13. Hermansky, H.: “Exploring temporal domain for robustness in speech recognition,” invited paper. Proceedings of the 15th International Congress on Acoustics, 3:61-64, 1995.
14. Language Identification Corpus of the Institute for Infocomm Research
15. Wang, H.-C.: MAT-a project to collect Mandarin speech data through networks in Taiwan. Int. J. Comput. Linguistics Chinese Language Process. 1 (2) (February 1997) 73-89.
16. http://cslu.cse.ogi.edu/corpora/corpCurrent.html
17. Campbell, W. M., Campbell, J. P., Reynolds, D. A., Jones, D. A. and Leek, T. R.: Phonetic speaker recognition with support vector machines. Proc NIPS, 2003

Contextual Maximum Entropy Model for Edit Disfluency Detection of Spontaneous Speech

Jui-Feng Yeh[1], Chung-Hsien Wu[2], and Wei-Yen Wu[2]

[1] Department of Computer Science and Information Engineering,
Far East University
No. 49, Chung-Hua Rd., Hsin-Shih, Tainan County 744
[2] Department of Computer Science and Information Engineering,
National Cheng Kung University,
No. 1, Ta-Hsueh Road, Tainan,
ralph@cc.feu.edu.tw, {chwu, wywu}@csie.ncku.edu.tw

Abstract. This study describes an approach to edit disfluency detection based on maximum entropy (ME) using contextual features for rich transcription of spontaneous speech. The contextual features contain word-level, chunk-level and sentence-level features for edit disfluency modeling. Due to the problem of data sparsity, word-level features are determined according to the taxonomy of the primary features of the words defined in Hownet. Chunk-level features are extracted based on mutual information of the words. Sentence-level feature are identified according to verbs and their corresponding features. The Improved Iterative Scaling (IIS) algorithm is employed to estimate the optimal weights in the maximum entropy models. Performance on edit disfluency detection and interruption point detection are conducted for evaluation. Experimental results show that the proposed method outperforms the DF-gram approach.

Keywords: Disfluency, maximum entropy, contextual feature, spontaneous speech.

1 Introduction

In the latest decade, the research of speech recognition has been significantly improved in practice. Current speech recognition systems output simply a stream of words for the oral reading speech. However, the variety of spontaneous speech acutely degrades the performances of speech recognition and spoken language understanding. Interactive spoken dialog systems face new challenges for speech recognition. One of the most critical problems in spoken dialog systems is the prevalence of speech disfluencies, such as hesitation, false starts, self-repairs, etc. Edit disfluency uttered by the speakers is a mortal factor for spoken language understanding and should be detected and corrected for better understanding of the utterance's meaning [1].

Edit disfluencies involve syntactically relevant content that is repeated, revised, or abandoned with the structural pattern composed of deletable region, interruption point, editing term (optional) and correction part. Deletable region is defined as the

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 578 – 589, 2006.

portion of the utterance that is corrected or abandoned. Interruption point is the position at which point the speaker break off the original utterance and fluent speech becomes disfluent. Editing term is composed of filled pause, discourse marker, and explicit editing term.

Edit disfluency are categorized as simple and complex edit disfluencies. Simple edit disfluencies are further divided into three categories, namely repetitions, revisions or repairs and restarts. Complex edit disfluecy represents that the corrected portion of one edit disfluency contains another disfluency and is composed of several simple edit disfluencies in the same sentence or utterance. Take the following fluent sentence as an example: 我明天想要去台北 (I want to go to Taipei tomorrow.). The definition and example corresponding to each category of edit disfluency are illustrated as follows:

(1) *Repetition*: the abandoned words repeated in the corrected portion of the utterance, as depicted in the following:

Chinese sentence : 我明天* 明天想要去台北
English translation: (I want to go to Taipei tomorrow * tomorrow.)
Correction : 我~~明天~~* 明天想要去台北

(2) *Revision or repair*: although similar to repetitions, the corrected portion that replaces the abandoned constituent modifies its meaning, rather than repeats it.

Chinese sentence: 我 今天* 明天想要去台北
English translation: (I want to go to Taipei today * tomorrow.)
Correction : 我~~今天~~* 明天想要去台北

(3) *Restarts or false starts*: a speaker abandons an utterance and neither corrects it nor repeats it partially or wholly, but instead restructures the utterance and starts over.

Chinese sentence: 我 * 我明天想要去台北
English translation: (I * I want to go to Taipei tomorrow.)
Correction : ~~我~~* 我明天想要去台北

where the dashed line “---” represents the corrected portion and the interruption point (IP) is marked with “*”.

1.1 Related Works

Much of the previous research on detecting edit disfluency has been investigated for improving performance of spoken language understanding. Coquoz found it is very useful for enriching speech recognition by processing edit disfluency especially for spontaneous speech recognition [2]. There are several approaches which can automatically generate sentence information useful to parsing algorithm [3] [4]. To identify and verify a speech act precisely, interference caused by speech repairs should be considered. Accordingly, a reliable model is desired to detect and correct conversation utterances with disfluency [5].

For edit disfluency detection, several cues exist to suggest when some edit disfluency may occur and can be observed from linguistic features[6] , acoustic features [7] or integrated knowledge sources [8]. Shriberg et al. [9] outlined the phonetic consequences of disfluency to improve disfluency processing methods in speech applications. Savova and Bachenko presented four disfluency detection rules using intonation, segment duration and pause duration [10]. IBM has adopted a discriminatively trained full covariance Gaussian system [11] for rich transcription. Kim et al. utilized a decision tree to detect the structural metadata [12]. Furui et al. [13] presented the corpus collection, analysis and annotation for conversational speech processing. Charniak and Johnson [14] proposed the architecture for parsing the transcribed speech using an edit word detector to remove edit words or fillers from the sentence string, and adopted a standard statistical parser to parse the remaining words. The statistical parser and the parameters estimated by boosting are employed to detect and correct the disfluency. Lease et al. later presented a novel parser to model the structural information of spoken language by Tree Adjoining Grammars (TAGs) [15-16]. Honal and Schultz used TAG channel model incorporating more syntactic information to achieve good performance in detecting edit disfluency [17]. The TAG transducer in the channel model is responsible for generating the words in the reparandum and the interregnum of a speech repair from the corresponding repair. Heeman et al. presented a statistical language model that can identify POS tags, discourse markers, speech repairs and intonational phrases [18-19]. By solving these problems simultaneously, the detection of edit disfluency was addressed separately. The noisy channel model was proposed to model the edit disfluency [16] [20] [21]. Snover et al. [22] integrated the lexical information and rules for disfluency detection using transformation-based learning. Hain et al. [23] presented techniques in front-end processing, acoustic modeling, language and pronunciation modeling for transcribing conversational telephone speech automatically. Soltau et al. transcribed telephone speech using LDA [24]. Harper et al. utilized parsing approaches to rich transcription [25]. Liu et al. not only detected the boundaries of sentence-like units using the conditional random fields [26], but also compared the performances of HMM, maximum entropy [27] and conditional random fields on disfluency detection [28].

1.2 Methods Proposed in This Paper

Berger et al. first applied the maximum entropy (ME) approach to natural language processing and achieved a better improvement compared to conventional approaches in machine translation [29]. Huang and Zweig proposed a maximum entropy model to deal with the sentence boundary detection problem [27]. Using the maximum entropy model to estimate conditional distributions provides a more principled way to combine a large number of overlapping features from several knowledge sources. In this paper, we propose the use of the maximum entropy approach with contextual features, containing word, chunk and sentence features, for edit disfluency detection and correction. Given labeled training data with edit disfluency information, maximum entropy is a statistical technique which predicts the probability of a label given the test data using the improved interactive scaling algorithm (IIS). The language related structural factors are taken as the features in the maximum entropy model.

1.3 Organization

The rest of this paper is organized as follows. Section 2 introduces the framework of the edit disfluency detection and correction using maximum entropy model and the language features adopted in this model. The weight estimation using improved iterative scaling algorithm are also described in this section. Section 3 summarizes the experimental results, along with a discussion made of those results. In Section 4, we conclude the findings and the directions for future work.

2 Contextual Maximum Entropy Models for Edit Disfluency Correction

In this paper, we regard the edit disfluency detection and correction as a word labeling problem. That is to say, we can identify the edit disfluency type and the portion category of every word in the utterance and accordingly correct the edit disfluency. Herein, the edit disfluency type includes normal, repetition, revision and restart. The portion categories contain sentence-like units, deletable region, editing term and correction part. Since the interruption point (IP) appears in an inter-word position that speaker breaks off the original utterance, all the inter-word positions in an utterance are regarded as the potential IP positions. Therefore, as long as the labels of the words in the utterance are determined, the detected edit disfluency can be corrected. Fig. 1 shows the example of an utterance with "**revision**" disfluency.

For edit disfluency detection, the maximum entropy model, also called log-linear Gibbs model, is adopted, which uses contextual features and takes the parametric exponential form for edit disfluency detection:

$$P\left(PT_{w_i} \mid W, F\right) = \frac{1}{Z_\lambda\left(W, F\right)} \exp\left(\sum_k \lambda_k f_k\left(PT_{w_i}, W, F\right)\right) \tag{1}$$

where PT_{w_i} contains the edit disfluency type and the portion category of word w_i. w_i denotes the *i-th* word in the speech utterance *W*. *F* is the feature set used in the contextual maximum entropy model. $f_k(\cdot)$ is an indicator function corresponding to contextual features described in the next section. λ_k denotes the weight of feature $f_k(\cdot)$. $Z_\lambda(W,F)$ is a normalizing factor calculated as follows:

$$Z_\lambda\left(W, F\right) = \sum_{PT_{w_i}} \exp\left(\sum_k \lambda_k f_k\left(PT_{w_i}, W, F\right)\right) \tag{2}$$

2.1 Contextual Features

Since edit disfluency is a structural pattern, each word cannot be treated independently for the detection of edit disfluency type and portion category. Instead, the features extracted from the contexts around the word w_i, called contextual

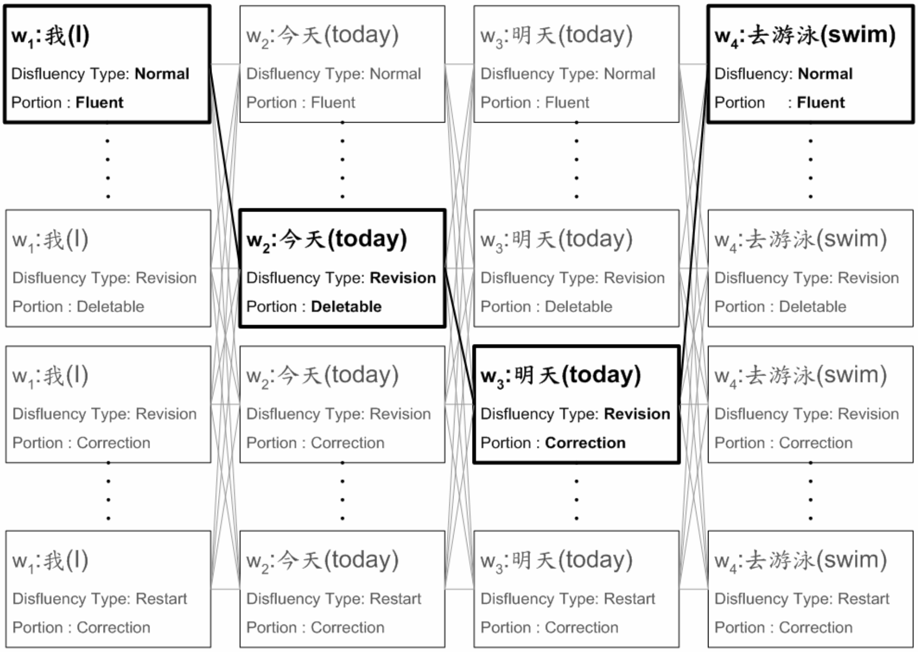


Fig. 1. The original utterance is "我今天*明天去游泳". The corrected sentence "我明天去游泳" can be obtained from the highlighted words with disfluency type and portion category, in which the word "今天"with "deletable" portion category is deleted.

features, should be considered for edit disfluency detection. The concepts derived from the primary features of the words defined in HowNet and co-occurrence of words are employed to form the contextual features for pattern modeling of edit disfluency. In order to consider the contextual information of a word, an observation window is adopted. The contextual features defined in this study are categorized bi-directional n-grams and variable-length structural patterns. These two features are described in the following.

(1) *Bi-directional n-gram features* are extracted from a sequence of words. Considering the words that follow and before the observed word w_i, the right hand side n-gram and left-hand side n-gram are obtained. Therefore, the uni-gram feature is shown as equation (3)

$$f_k\left(PT_{w_i},W,F_o\right)=\begin{cases}1 & if\ \ Class\left(w_i\right)=category_j\\ 0 & \qquad otherwise\end{cases} \tag{3}$$

Where $category_j$ denotes the *j-th* taxonomy defined in HowNet. The right hand side n-gram and left hand side n-gram are shown in equations (4) and (5), respectively.

$$f_k\left(PT_{w_i},W,F_R\right)=\begin{cases}1 & \text{if } Class\left(w_{i-n+1}\right)=category_{j_{i+n-1}} \wedge \cdots Class\left(w_i\right)=category_{j_i} \\ 0 & \text{otherwise}\end{cases} \tag{4}$$

$$f_k\left(PT_{w_i},W,F_L\right)=\begin{cases}1 & \text{if } Class\left(w_i\right)=category_{j_i} \wedge \cdots Class\left(w_{i+n-1}\right)=category_{j_{i+n-1}} \\ 0 & \text{otherwise}\end{cases} \tag{5}$$

The contextual feature set, employed in the proposed model, consists of uni-gram, right-hand side bi-gram, left-hand side bi-gram, right-hand side tri-gram, left-hand side tri-gram and right-hand side and left-and side n-grams. Editing terms containing fillers, discourse markers and negative words play important roles in edit disfluency detection using contextual features.

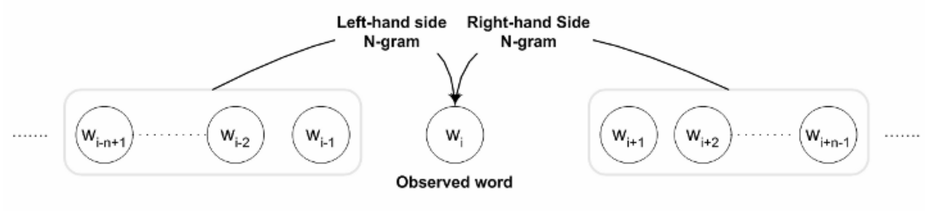


Fig. 2. Illustration of left-hand side n-gram and right-hand side n-gram contextual features

(2) *Variable-length structural patterns* are derived according to the characteristics of edit disfluencies. Since each word models only local information, structural information, such as sentences and phrases can be employed for syntactic pattern modeling. The units with variable length are considered to form the syntactic patterns using the sentences and chunks as the building blocks instead of words only. That is to say, we can extend the contextual scope by sentence and chunk n-grams to obtain better resolution of edit disfluency as shown in Fig. 3.

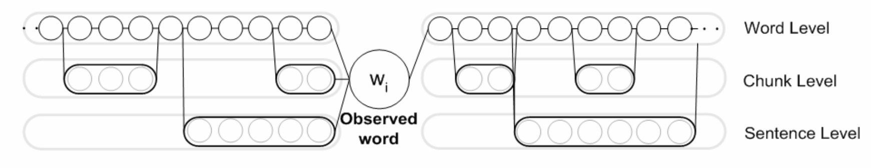


Fig. 3. Illustration of the syntactic patterns with three kinds of units: word, chunk and sentence

The sentence-level feature is identified according to the verbs and their corresponding necessary arguments defined in [30]. The chunk-level feature is extracted by the mutual information of the word sequence $c_i c_j$ according to co-occurrence and term frequencies of c_i and c_j.

$$Chunck\left(c_i c_j\right) \equiv I\left(\log_2 \frac{P\left(c_i c_j\right)}{P\left(c_i\right)P\left(c_j\right)} \geq \xi\right) \tag{6}$$

Where $Chunck(\cdot)$ denotes the function to determine if the word sequence is a chunk. c_i and c_j can be a word or a chunk. $I(\cdot)$ and ξ are the indicator function and the threshold of mutual information, respectively.

2.2 Parameter Estimation

In maximum entropy modeling, improved iterative scaling algorithm (IIS) is employed to estimate the parameter λ_k. The weight vector $\Lambda \equiv \{\lambda_1, \lambda_2, \cdots, \lambda_n\}$ is updated iteratively using IIS with the constraint that the expected values of various feature functions match the empirical averages in the training data. That is to say, the conditional log likelihood is also maximized over the training data. IIS algorithm is illustrated in Fig. 4.

```
Algorithm Improved iterative scaling algorithm (IIS)
Initial $\Lambda^{(0)} = (0, 0, \cdots, 0)^T$
Do
    Solve $\delta_i^{(t)}$ according to $E_{\tilde{p}}(f_i) = \sum_x \tilde{p}(x) \exp\left(\delta_i^{(t)} \sum_i f_i(x)\right) \times f_i(x)$
    $\Lambda^{(t+1)} = \Lambda^{(t)} + \delta^{(t)}$
Until converge
```

Fig. 4. The Improved iterative scaling algorithm (IIS) algorithm for estimating the weight vector

Where $E_{\tilde{p}}$ is the expectation operator with respect to the empirical distribution. $\delta^{(t)} \equiv \left\{\Delta\lambda_1^{(t)}, \Delta\lambda_2^{(t)}, \cdots, \Delta\lambda_n^{(t)}\right\}$ represents the increment of weight vector $\Lambda^{(t)}$ for the *t-th* iteration.

3 Experiments

3.1 Data Preparation

The Mandarin Conversational Dialogue Corpus (MCDC) [31], collected from 2000 to 2001 at the Institute of Linguistics of Academia Sinica, Taiwan, comprising 30 digitized conversational dialogues numbered from 01 to 30 of a total length of 27 hours, is used for edit disfluency detection and correction in this paper. The annotations described in [31] give concise explanations and detailed operational definitions of each tag. Like SimpleMDE, direct repetitions, partial repetitions, overt repairs and abandoned utterances are considered as the edit disfluency and the related information are labeled in MCDC.

Besides the subsyllable acoustic models, filler models [32] and discourse markers were defined for training using the Hidden Markov Model Toolkit (HTK) [33], and the recognized results were considered in language modeling. A speech recognition engine using HTK was built for syllable recognition using eight states (three states for the Initial part, and five states for the Final part in Mandarin syllable). The input speech was pre-emphasized with a coefficient of 0.97. A frame size of 32 ms (512 samples) with a frame shift of 10.625 ms (170 samples) was used. The MAT Speech Database, TCC-300 [34] and MCDC were used to train the parameters in the speech recognizer.

3.2 Experiments on Edit Disfluency Detection

Edit word detection (EWD) detects the input speech containing the words in the deletable regions. One of the primary metrics for the evaluation of edit disfluency correction is the edit word detection rate defined in RT'04F. This method is defined as the average number of missed edit word detections and falsely detected edit words per reference IP:

$$Error_{EWD} = \frac{n_{M-EWD} + n_{FA-EWD}}{n_{EWD}} \tag{7}$$

where n_{M-EWD} is the number of deletable edit words in the reference transcription that are not covered by the deletable regions of the system-detected edits; n_{FA-EWD} denotes the number of reference words that are not deletable, yet are covered by deletable regions of the system-detected edits, and n_{EWD} represents the number of deletable reference edit words. For assessing the performance of the proposed model, the statistical language model for speech disfluencies, proposed by Stolcke and Shriberg called DF-gram [35], is developed for comparison. The model is based on a generalization of the standard N-gram language model. The dynamic programming is used to compute the probability of a word sequence considering possible hidden disfluency events. Table 1 presents the results of the proposed method and DF-gram.

Table 1. Results of the proposed maximum entropy model and DF-gram

	Missed	False Alarm	Error ($Error_{EWD}$)
Maximum Entropy	0.05	0.20	0.25
DF-gram	0.13	0.16	0.29

The missed and false alarm error rates for the proposed maximum entropy approach are 0.05 and 0.2 respectively. The proposed contextual maximum entropy approach outperforms the DF-gram, especially for missed edit word errors. There are two reasons leading to disappointing results in false alarm: the insertion error of speech recognition and the misclassification of restart. These results indicate that the proposed model can handle repetition and revision disfluencies very well. However, it did not perform as well as expected for "**restart**" detection, where the improvement was less pronounced than that for other edit disfluency categories.

3.3 Experiments on Contextual Features

Since the feature set plays an important role in maximum entropy model, this experiment is designed for obtaining optimum size of the observation window. The right-hand side and left-hand side contextual features are selected symmetrically to form the feature set used in the proposed model. According to the number of units within the observation window, we can obtain the n-gram features based on words chunks and sentences. For example, if the number of words within the observation window is one, the feature set contains only the uni-gram feature. The determination of edit word depends only on the word itself in the observation window. If the observation window size is five, the right-hand side bi-gram, right-hand side tri-gram, left-hand side bi-gram, left-hand side tri-gram and uni-gram are included in the feature set. Table 2 shows the results for edit word detection with various observation window sizes.

Table 2. Edit word detection results for various observation window sizes to form the feature set. $\text{Error}_{\text{EWD(I)}}$ and $\text{Error}_{\text{EWD(O)}}$ represent the error rates for inside and outside tests, respectively.

Observation Window Size	1	3	5	7	9	11
$\text{Error}_{\text{EWD(I)}}$	0.158	0.143	0.117	**0.111**	0.114	0.122
$\text{Error}_{\text{EWD(O)}}$	0.201	0.222	0.209	**0.190**	0.196	0.197

Compared to observation window size of three, the feature set from that is one can provide comparable performance. The best result appears when the observation window size is seven. The performance of edit word detection task gradually declines as the observation window size increases. The reason is that the abandoned deteletable region usually contains few words. According to our observation, another finding of this experiment is the unit using sentence can provide significant improvement of the resolution between "restart" and fluent sentence. For example the fluent sentence "我聽說您去台北 (I heard that you want to go to Taipei)" is confused with the disfluent sentence "我去 您去台北 (I go * you go to Taipei)" when the model with the word-based bi-directional n-gram features. By introducing sentence-level feature, these two sentences can be regarded as "我聽說 [S] (I heard [S])" and "我去 [S] (I go * [S])". The verb "聽說 (heard)" can be followed by a sentence [S], while the verb "去 (go)" should be followed by a noun. Considering the characteristics of verbs and sentence structural information, we can achieve significant improvement on detecting the "restart" disfluency.

3.4 Results Analysis on Corresponding Edit Disfluency Types

As we have observed, there are different effects due to various edit disfluency types. For comparison, we also show the detection results of repletion, repair and restart by the proposed maximum entropy model and DF-gram in Table 3.

Table 3. The results of different edit disfluency types in edit word error by the proposed maximum entropy model and DF-gram approaches

	Repetition	Repair	Restart	$\text{Error}_{\text{EWD}}$
ME	0.12	0.24	0.29	0.25
DF-gram	0.15	0.28	0.34	0.29

The result of maximum entropy model is better than that of DF-gram for three kinds of edit disfluencies especially the “restart”. In fact, “restart” is usually confused with two cascaded normal sentences. The performance of “restart” detection is improved significantly by introducing the features of chunk and sentence features. In addition, the number of verbs within contextual scope is also helpful to detect “restart”.

4 Conclusion and Future Work

This paper has presented a contextual maximum entropy model to detect and correct edit disfluency that appears in spontaneous speech. The contextual features of variable length are introduced for modeling contextual patterns that contain deletable region, interruption point, editing term and correction part. Improved iterative scaling algorithm is used to estimate the weight of the proposed model. According to the experimental results, we can find the proposed method can achieve an error rate of 25% in edit word detection. Besides the word-level features, chunk-level and sentence-level features are adopted as the basic units to extend the contextual scope for capturing not only local information but also structural information. The results show that the proposed method outperforms the DF-gram.

For the future work, prosodic features are also beneficial for interruption point and edit disfluency detection. In addition, tagging training data is labor intensive and bias due to personal training, automatic or semi-automatic annotation tools should be developed to help the transcription of dialogs or meeting records.

References

1. Nakatani, C. and Hirschberg, J.: A speech-first model for repair detection and correction. Proceedings of the 31 Annual Meeting of the Association for Computational Linguistics, (1993) 46-53.
2. Coquoz, S.: Broadcast News segmentation using MDE and STT Information to Improve Speech Recognition. International Computer Science Instute, Tech. Report., (2004).

3. Gregory, M., Johnson, M. and Charniak, E.: Sentence-internal prosody does not help parsing the way punctuation does not Help Parsing the Way Punctuation Does, Proc. NAACL, (2004). 81-88.
4. Kahn, J.G., M. Ostendorf and C. Chelba: Parsing Conversational Speech Using Enhanced Segmentation. Proc. HLT-NAACL, 2004. pp. 125-128.
5. Wu, C.-H. and Yan, G.-L.: Speech Act Modeling and Verification of Spontaneous Speech With Disfluency in a Spoken Dialogue System. IEEE Transaction on Speech and Acoustic Processing, Vol. 13, No. 3, (2005). 330-344.
6. Yeh J.-F. and Wu C.-H.: Edit Disfluency Detection and Correction Using a Cleanup Language Model and an Alignment Model. IEEE Transactions on Audio, Speech, and Language Processing, Vol. 18, Issue 5,(2006), 1574-1583.
7. Shriberg, E., Stolcke, A., Hakkani-Tur, D. and Tur, G.: Prosody-based automatic segmentation of speech into sentences and topics", Speech Communication, 32(1-2), (2000), 127-154.
8. Bear, J., Dowding, J. and Shriberg, E.: Integrating multiple knowledge sources for detecting and correction of repairs in human computer dialog. Proc. of ACL, (1992). 56–63.
9. Shriberg, E., Ferrer, L., Kajarekar, S., Venkataraman, A. and Stolcke, A.: Modeling prosodic feature sequences for speaker recognition. Speech Communication 46 , (2005), 455–472
10. Savova, G. and Bachenko, J.: Prosodic features of four types of disfluencies. in Proc. of DiSS 2003, (2003), 91–94.
11. Soltau, H., Kingsbury, B., Mangu, L., Povey, D., Saon, G., and Zweig, D.: The IBM 2004 Conversational Telephony System for Rich Transcription. In Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005 (ICASSP '05). (2005), 205-208.
12. Kim, J., Schwarm, S. E. and Ostendorf, M.: Detecting structural metadata with decision trees and transformation-based learning. Proceedings of HLT/NAACL 2004, (2004), 137–144.
13. Furui, S., Nakamura, M., Ichiba, T. and Iwano, K.:Analysis and recognition of spontaneous speech using Corpus of Spontaneous Japanese. Speech Communication 47, (2005), 208–219
14. Charniak, E. and Johnson, M.: Edit detection and parsing for transcribed speech. in Proc. of NAACL 2001, (2001), 118–126.
15. Johnson, M. and Charniak, E.: A TAG-based noisy channel model of speech repairs. in Proc. of ACL 2004, (2004). 33-39.
16. Lease, M., Charniak, E., and Johnson, M.: Parsing and its applications for conversational speech, in Proc. of ICASSP 2005, (2005).
17. Honal, M. and Schultz T.: Automatic Disfluency Removal on Recognized Spontaneous Speech - Rapid Adaptation to Speaker Dependent Disfluencies. In Proceedings of ICASSP '05. (2005), 969-972.
18. Heeman, P. A. and Allen, J.: Speech repairs, intonational phrases and discourse markers: Modeling speakers' utterances in spoken dialogue. Computational Linguistics, vol. 25, (1999), 527–571.
19. Heeman, P.A., Loken-Kim, K., Allen, J.: Combining the detection and correction of speech repairs. In Proceedings of the 4rd International Conference on Spoken Language Processing (ICSLP-96). (1996), 358--361.

20. Honal, M. and Schultz, T.: Automatic disfluency removal on recognized spontaneous speech - rapid adaptation to speaker dependent dislfuencies. in Proc. of ICASSP 2005, (2005). 969-972
21. Honal, M. and Schultz, T. "Corrections of disfluencies in spontaneous speech using a noisy-channel approach," in Proc. of Eurospeech, 2003, (2003). 2781-2784.
22. Snover, M., Dorr, B., and Schwartz, R.: A lexically-driven algorithm for disfluency detection. in Proc. of HLT/NAACL 2004, (2004), 157-160.
23. Hain, T., Woodland, P. C., Evermann, G., Gales, M.J.F., Liu, X., Moore, G. L., Povey, D. and Wang, L.: Automatic Transcription of Conversational Telephone Speech. IEEE Transactions on Speech and Audio Processing: Accepted for future publication.
24. Soltau, H., Yu, H., Metze, F., Fugen, C., Qin, J., Jou, S.-C.: The 2003 ISL rich transcription system for conversational telephony speech. In Proceedings of Acoustics, Speech, and Signal Processing 2004 (ICASSP), (2004), 17-21.
25. Harper, M., Dorr, B. J., Hale, J., Roark, B., Shafran, I., Lease, M., Liu, Y., Snover, M., Yung, L., Krasnyanskaya, A. and Stewart, R.: Final Report on Parsing and Spoken Structural Event Detection, Johns Hopkins Summer Workshop, (2005).
26. Liu, Y., Stolcke, A., Shriberg, E. and Harper, M.: Using Conditional Random Fields for Sentence Boundary Detection in Speech. In Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics: ACL 2005, (2005)
27. Huang J. and Zweig, G.: Maximum Entropy Model for Punctuation Annotation from Speech. In Proceedings of ICSLP 2002, (2002). 917-920.
28. Liu, Y., Shriberg, E., Stolcke A. and Harper, M.: Comparing HMM, Maximum Entropy, and Conditional Random Fields for Disfluency Detection. in Proc. of Eurospeech, 2005, (2005). 3313-3316.
29. Berger, A. L., Pietra, S. A. D. and Pietra V. J. D.: A maximum entropy approach to natural language processing. Computational Linguistics, Vol. 22, (1996). 39-72.
30. Chinese Knowledge Information Processing Group (CKIP): Technical Report 93-05: Chinese Part-of-speech Analysis. Academia Sinica, Taipei. (1993).
31. Tseng, S.-C. and Liu, Y.-F.: Annotation of Mandarin Conversational Dialogue Corpus. CKIP Technical Report no. 02-01." Academia Sinica. (2002).
32. Wu, C.-H. and Yan, G.-L.: Acoustic Feature Analysis and Discriminative Modeling of Filled Pauses for Spontaneous Speech Recognition, Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology, 36, (2004), 87-99.
33. Young, S. J., Evermann, G., Hain, T., Kershaw, D., Moore, G. L., Odell, J. J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P. C.: The HTK Book. Cambridge, U.K.: Cambridge Univ. Press, (2003).
34. MAT Speech Database – TCC-300 (http://rocling.iis.sinica.edu.tw/ROCLING/MAT/Tcc_300brief.htm)
35. Stolcke, A. and Shriberg, E.: Statistical Language Modeling for Speech Disfluencies, In Proceedings of ICASSP-96, (1996), 405-408.

Automatic Detection of Tone Mispronunciation in Mandarin

Li Zhang[1,2,⋆], Chao Huang[2], Min Chu[2], Frank Soong[2],
Xianda Zhang[1], and Yudong Chen[2,3]

[1] Tsinghua University, Beijing, 100084
[2] Microsoft Research Asia, Beijing, 100080
[3] Communication University of China, 100024
zhang-li04@mails.tsinghua.edu.cn,
{chaoh, minchu, frankkps}@microsoft.com, zxd-dau@tsinghua.edu.cn

Abstract. In this paper we present our study on detecting tone mispronunciations in Mandarin. Both template and HMM approaches are investigated. Schematic templates of pitch contours are shown to be impractical due to their larger pitch range of inter-, even intra-speaker variation. The statistical Hidden Markov Models (HMM) is used to generate a Goodness of Pronunciation (GOP) score for detection with an optimized threshold. To deal with the discontinuity issue of the F0 in speech, the multi-space distribution (MSD) modeling is used for building corresponding HMMs. Under an MSD-HMM framework, detection performance of different choices of features, HMM types and GOP measures are evaluated.

1 Introduction

During the past few years, much progress has been made in the area of computer-assisted language learning (CALL) systems for nonnative language learning. Automatic speech recognition (ASR) technology by defining a proper Goodness of Pronunciation (GOP) measure is applied to grade the pronunciation quality at both speaker and sentence levels [4][5][6]. To improve the feedback quality, precise knowledge of the mispronunciation is required. Detection of mispronunciation in a speaker's utterance have been developed and achieved good performance[7][8][9].

CALL systems for Mandarin are also desired. What is more,the proficiency test of Mandarin for Chinese (PSC) becomes more and more popular recently. Phonetic experts are required during the test as judgers, which makes the test costly, time-consuming and not absolutely objective. So automatic assessment of Mandarin is very necessary.

As we know, Mandarin is a tonal language. Compared with finals and initials, tones are more difficult to be pronounced correctly because they are much easily influenced by the dialect of a speaker. In PSC, the goodness of tone pronunciation is one of the most important factors to be used to grade the testing speaker.

⋆ Join this work as a visiting student at MSRA.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 590–601, 2006.

However, in previous works [1][2][3], assessment of tone pronunciation has not been taken into consideration. In addition, studies before mostly focused on rating in the sentence level and speaker level. In contrast to existing works, we focus on the automatic detection of tone mispronunciation.

In this paper, some analysis of tone mispronunciations occurred in PSC is first given in Section 2. Based on the analysis, two kinds of detection methods are proposed: template based method and statistical models. Section 3 describes the template methods. Section 4 investigates the statistical methods based on multi-space distribution Hidden Markov Models (MSD-HMM) under different setups such as model types, feature combinations and GOP measures. Section 5 demonstrates the experiment results and analysis. Conclusions are given in Section 6.

2 Some Analysis About Tone Mispronunciations in PSC

There are five tones in Mandarin Chinese and first four tones are widely used. Although many factors affect the characteristics of tones, pitch is considered as the most discriminative factor. In phonetics theory, pitch range is usually divided into 5 levels, naming 1 to 5 and level 5 corresponding to the highest. Traditional description of Mandarin tones is shown in Table 1.

Table 1. Tones of Mandarin

Chinese Pronunciation	Tone Symbol	Tone Description
Tone1	55	high level
Tone2	35	high rising
Tone3	214	low falling
Tone4	51	high falling

According to the experts' experience in PSC, rules to evaluate tone pronunciations are also based on the 5-level description. General mispronunciations of tone are discussed below. Typical examples with tone mispronunciations in our database are plotted in Figure 1. Pitch is extracted by the entropic signal processing system (ESPS).

The first tone (55): it is sometimes pronounced unevenly as shown in Figure 1a. Another representative mistake is that pitch values are not high enough, such as 44 or even lower value. This kind of mistakes always exist in the pronunciations of people living in south-north of China.

The second tone (35): it is easily confused with the third tone when the descending part of the contour lasts long enough to be perceived, as shown in Figure 1b.

The third tone (214): a rising trend is required at the end of the pitch contour for isolated syllables, which is not strict in the continuous speech. Some speakers ignore such requirement and pronounce as 21 (Figure 1c). Some speakers pronounce it as 35 that becomes the second tone (Figure 1d). Another mistake

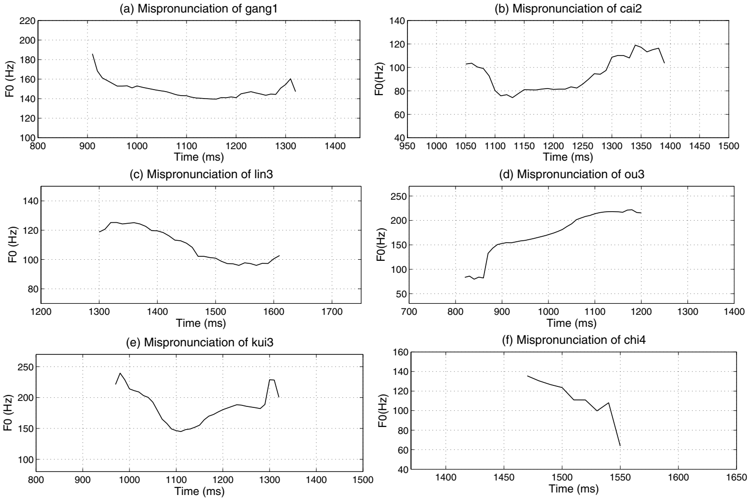


Fig. 1. F0 contours of examples with tone mispronunciations

is beginning value is so high to be comparable with or even higher than the ending's, such as 313 or 413 and so on. Figure 1e is an example of 413.

The fourth tone (51): the beginning is sometimes pronounced not high enough. Speakers whose pitch ranges are not wide enough are more possible to make such mistakes, as shown in Figure 1f.

According to these analysis, we present two kinds of methods for the detection of tone mispronunciations.

3 Automatic Detection Methods Based on Template

The expert's judgement agree with most of results from analysis of pitch contours in our experiments. It seems that just analyzing pitch contours can make the detection. The third tone is more prone to be pronounced incorrectly and it covers more than 80% mistakes or defectiveness occurred in our database. In addition, there are lots of variant mispronunciations for tone 3. Therefore, we will use tone 3 as the studying case to investigate the template method.

3.1 Methods Based on Five-Level Description

The most typical mispronunciation of tone 3 are 21, 35, 313, 413 and so on. A method to detect these mispronunciations is firstly partitioning the testing speaker's pitch range into five levels according to his pitch contours; then describing his pitches using them. If the description is not 214, it is judged to be incorrect.

This method seems reasonable and very simple. However, it is very difficult to partition the pitch range into five levels, for they are just relative and regions of neighbor levels always have overlap. Figure 2 shows range of 1st and 5th levels of a male speaker, who speaks Mandarin very well. Surprisingly, the 5th level decided by range of the first tone (55) even have an overlap with the 1st level decided by the ending values of the fourth tone (51) in this case. Range within a level are also very wide. These reasons make the partition too difficult to be implemented.

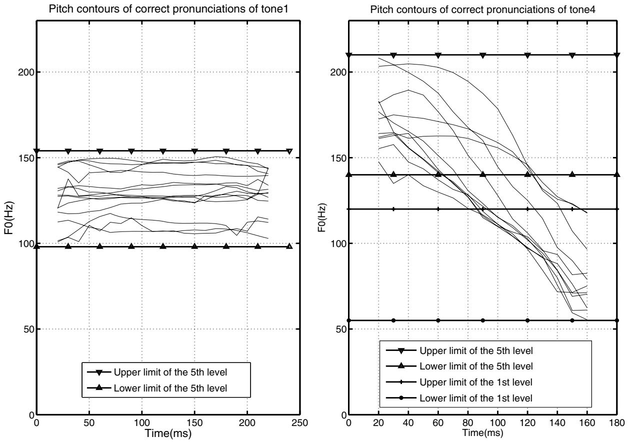


Fig. 2. Examples of pitch contour for tone1 and tone4 from a single speaker

3.2 Methods Based on Pitch Value Comparison

Considering the fact of existing big overlap among 5-value quantization of pitch range as shown in last sub-section, we can also check the relative value of pitch for a given tone contour as shown in Figure 3. For example, we can detect the mispronunciations like 21 and 35 by just counting the durations of ascending and descending parts of a contour. Through comparing the values of beginning and ending parts, we can detect the mistakes such as 313,314 and so on.

However, it is also not practical: firstly, there are usually elbows at the beginning and ending of pitch contours that make the detection of the beginning and ending segment unreliable. Duration estimation based on segmentation results becomes more unreliable; secondly, there are too many threshold parameters to be predefined or tuned and some of them are even beyond expert's perceiving resolution.

Comparing with the template methods above, there are fewer threshold to be predefined in statistical methods such as HMMs and more features combinations in addition to pitch can be applied flexibly in such framework.

4 Automatic Detection Methods Based on Statistic Models

In this section statistical methods based on HMMs are discussed. A brief introduction is given first: syllables pronounced by golden speakers are used to

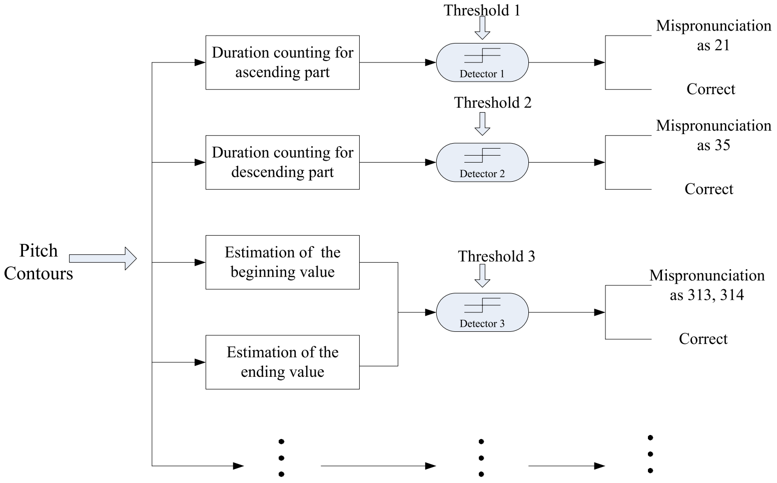

Pitch Contours
Duration counting for ascending part
Duration counting for descending part
Estimation of the beginning value
Estimation of the ending value
Threshold 1
Threshold 2
Threshold 3
Detector 1
Detector 2
Detector 3
Mispronunciation as 21
Correct
Mispronunciation as 35
Correct
Mispronunciation as 313, 314
Correct

Fig. 3. Flow chart for the template method

train the speaker independent HMMs and phone set [10] is used, in which each tonal syllable is decomposed into an initial and a tonal final. After forced alignment, GOP scores are computed within the segment. Finally, with the help of a threshold, detection is operated. Detailed descriptions about choice of model types, feature vectors and GOP measures are given in the following subsections.

4.1 Experimental Corpora

The total corpora of the experiment contain 100 speakers, each speaker reads 1267 syllables. Among them we chose 80 speakers (38 male and 42 female) as the training set. The rest 20 speakers(10 female and 10 male) are used for testing. Randomly 100 syllables from each of these 20 speakers and totally 2000 syllables are selected for expert's evaluation. Expert with national level scores each pronunciation correct or not and points out where the problem is, such as tone mispronunciation if there is.

4.2 Choice of Models

Studies have indicated that F0 related features can greatly improve tone recognition accuracy. For F0 there is no observation in the unvoiced region and some methods have been proposed to deal with it[11][12][14]. Among them, multi-space distribution (MSD) approach, first proposed by Tokuda [13] for speech synthesis, have also achieved good performance in tonal language speech recognition[14]. The MSD assumed that the observation space can made of multiple subspaces with different priors and different distribution forms (discrete or continuous pdf)can be adopted for each subspace. We have adopted two kinds of models to solve the problem of discontinuity of F0 feature in speech: MSD models in which the observation space are modeled with two probability spaces, discrete one for unvoiced part and continuous one (pdf) for voiced F0 contour, and the other model in which random noise is interpolated to the unvoice region to keep the F0 related feature continuous during the whole speech. Their performance in terms of tone recognition error rate are compared in our experiments.

Acoustic feature vector contains 39-dimensional spectral features (MFCC-E-D-A-Z), and 5-dimensional F0 related features, consisting of logarithmic F0, its first, second order derivatives, pitch duration and long-span pitch[15]. In MSD models, we separate the features into two streams: one is MFCC-E-D-A-Z, the other is 5 dimensional F0 related features and only one stream of 44-dimension is used in conventional models.

We compare their performance in Table 2. MSD models perform better than conventional models in both monophone and triphone cases, so MSD models are more proper for the detection of tone mispronunciation. In addition, tied state triphone models are better than monophone models in terms of recognition. We also compare their performance in the detection.

Table 2. Comparison of tone recognition error rate between conventional HMMs and MSD HMMs

Model Type	Model Size	Tone Error Rate (%)
HMMs, monophones	187*3(states), 16(mixtures/state)	18.05
MSD-HMMs, monophones	187*3(states), 16(mixtures/state)	8.85
HMMs, triphones	1500(tied states), 16(mixtures/state)	15.85
MSD-HMMs, triphones	1506(tied states), 16(mixtures/state)	7.81

4.3 Choice of Features

Spectral features such as MFCC can improve tone recognition accuracy in ASR systems. However, the most discriminative feature for tone is F0. To check whether MFCC are beneficial to the detection of tone mispronunciation, we use two kinds of feature vectors: F0 related features and its combination with MFCC-E-D-A-Z in our experiments.

Pitch ranges vary greatly for speakers and normalization is needed. Two normalization are proposed: pitch value is divided by the average of nonzero values within a syllable and logarithm of F0.

4.4 Choice of Goodness of Pronunciation Measures

We compare three types of scores for tone pronunciation: recognition scores, log-likelihood scores and log-posterior probability scores, all of which are computed within the HMM paradigm.

Recognition Scores. A simple measure for tone mispronunciation is just based on tone recognition results. If the tone is recognized correctly, its pronunciation is judged as correctand otherwise it will be judged as one with mistakes or defectiveness. Such kind of measure will be highly dependent on the pronunciation quality of the training data that used to generate the HMM models. For example, if one speaker mistakenly pronounce A to B and such case are observed a lot in training data and the decoding result may be still correct even for the wrong pronunciations.

Log-likelihood Scores. The log-likelihood score for each tonal segment is defined as:

$$l_i = \frac{1}{d_i} \cdot \sum_{t=t_i}^{t_i+d_i-1} log(p(y_t|tone_i)) \tag{1}$$

where,

$$p(y_t|tone_i) = \sum_{j=1}^{J_i} p(y_t|final_j, tone_i)P(final_j|tone_i) \tag{2}$$

$p(y_t|final_j, tone_i)$ is the likelihood of the current frame with the observation vector y_t. $P(final_j|tone_i)$ represents the prior probability of the $final_j$ given that its corresponding tone is $tone_i$. d is the duration in frames of the tonal final, t_0 is its starting frame index, and J_i is the total number of the final phones with $tone_i$. Normalization by the number of frames in the segment eliminates the dependency on the duration.

Log-posterior Probability Scores. Log-posterior probability scores perform better than log-likelihood scores in most CALL systems[6]. We also modify its formula in our case.

First, for each frame within the segment corresponding to $tone_i$, the frame-based posterior probability $p(tone_i|y_t)$ of $tone_i$ given the observation vector y_t is computed as follows:

$$\begin{aligned} p(tone_i|y_t) &= \frac{p(y_t|tone_i)P(tone_i)}{\sum_{m=1}^{4} p(y_t|tone_m)P(tone_m)} && (3) \\ &= \frac{\sum_{j=1}^{J_i} p(y_t|final_j, tone_i)P(final_j|tone_i)P(tone_i)}{\sum_{m=1}^{4} \sum_{j=1}^{J_m} p(y_t|final_j, tone_m)P(final_j|tone_m)P(tone_m)} && (4) \end{aligned}$$

$P(tone_m)$ represents the prior probability for $tone_m$. Then, the log-posterior probability for the $tone_i$ segment is defined as:

$$\rho_i = \frac{1}{d_i} \cdot \sum_{t=t_i}^{t_i+d_i-1} log(p(tone_i|y_t)) \tag{5}$$

5 Experiments and Results

The training and testing database for detection are the same as Section 4.1. MSD method is used to model golden pronunciations of tone. Performance of different model types, feature vectors and GOP measures are evaluated.

5.1 Performance Measurement

To evaluate performance of different setups, four decision types can be defined:

- Correct Acceptance(CA): A tone has been pronounced correctly and was detected to be correct;
- False Acceptance(FA): A tone has been pronounced incorrectly and was detected to be correct;
- Correct Rejection(CR): A tone has been pronounced incorrectly and was detected to be incorrect;
- False Rejection(FR): A tone has been pronounced correctly and was detected to be incorrect.

Given a threshold, all these decision types can be computed. Scoring accuracy (SA) defined as $CA + CR$ is always plotted as a function of FA for a range of thresholds to evaluate the performance of a detection system. The SA-FA curves are plotted for different setups in next sections.

5.2 Results for Different Model Types

Basic setups and tone recognition error rate of all models in our experiments are listed in Table 3.

Table 3. Tone recognition error rate of different model sets

Model Type	Feature Vector	Tone Error Rate (%)
Monophones	LogF0 (5)	11.20
Monophones	MFCC-E-D-A-Z, LogF0(5)	8.85
Monophones	MFCC-E-D-A-Z, Normalized F0(5)	7.75
Triphones	LogF0 (5)	9.25
Triphones	MFCC-E-D-A-Z, LogF0(5)	7.10
Triphones	MFCC-E-D-A-Z, Normalized F0(5)	6.05

Firstly we compared the performance of monophone and tied state triphone models. In all the experiments below, logF0 related 5 dimensional features mentioned are the same as [15] and we use "LogF0 (5)" for short. Table 3 shows that triphone models always perform better than monophone models for tone recognition. It is obvious that context information modeled by triphones is helpful for recognition.

Performance of these models on detection are shown in Figure 4. Log-posterior probability scores are chosen as the GOP measure. Triphone models achieve a little better performance than monophone models for the detection. Triphones model the context between initial and final in our case and tone also has a relationship with initial, which may be a reason why triphone models are beneficial for tone detection. However, the affection of initial on tone is very limited compared with final and it is why the advantage is not so much.

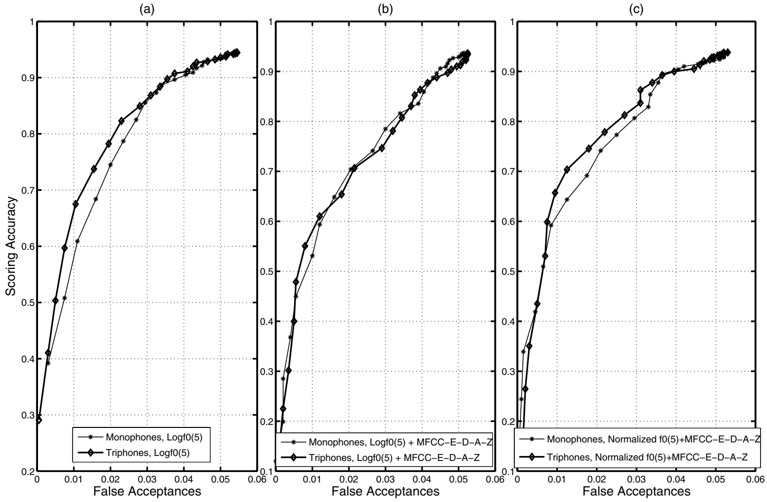


Fig. 4. Scoring accuracy versus false acceptance for monophones and triphones

5.3 Results for Different Features

In this section, we evaluate performance of different features. The basic setups and tone recognition error rate of the models are in Table 3. Log-posterior probability scores are still used as the GOP measures.

Figure 5 indicates that spectral features such as MFCC are useful for tone recognition as shown in Table 3,however, they seem not beneficial for the detection. The reference of detection is provided by the phonetic experts. We inferred that phonetic experts judge the quality of tone pronunciation probably mainly based on pitch contours as discussed in Section 2 and they are consistent with F0 related features. It is why MFCC features are probably of little helpful to improve the agreement.

Table 3 and Figure 6 show that normalization of F0 is more effective than logarithm for both recognition and detection. The reason is that normalization can reduce the pitch values into smaller range than logarithm, which makes models more independent of speakers.

5.4 Results for Different GOP Measures

GOP measure is another key factor for detection. In this section, we compare the performance of three different measures mentioned in Section 4.3.

Figure 7 shows log-posterior probability scores achieve the best performance among these measures. The assumption behind using posterior scores is that the better a speaker has pronounced a tone, the more likely the tone will be over the remain tones. The experiment results indicates such assumption is reasonable.

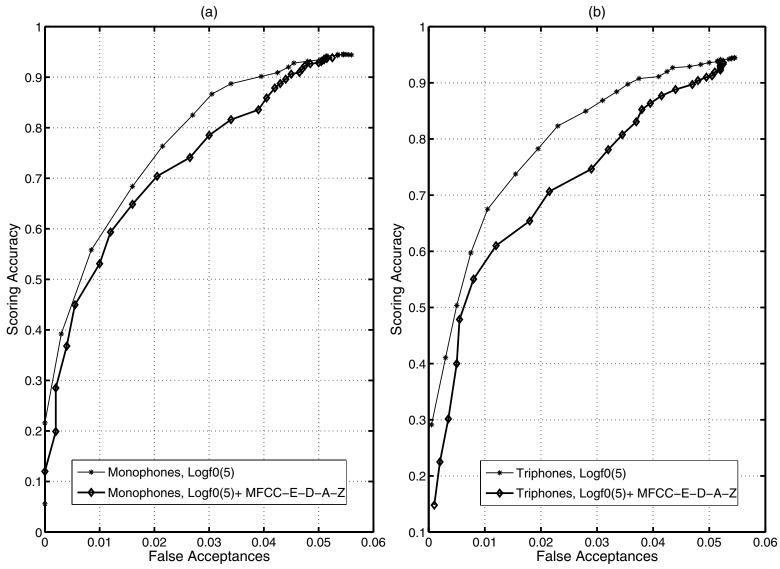


Fig. 5. Scoring accuracy versus false acceptance for different features

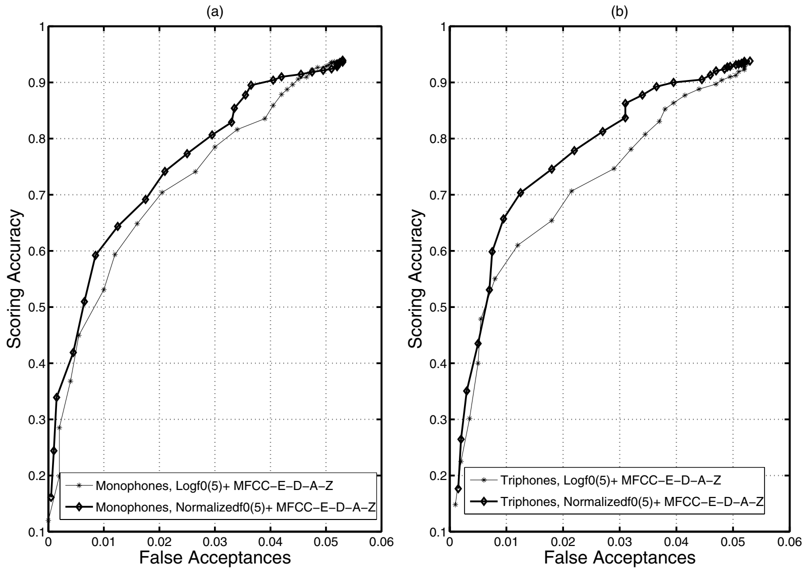


Fig. 6. Scoring accuracy versus false acceptance for different processing of F0

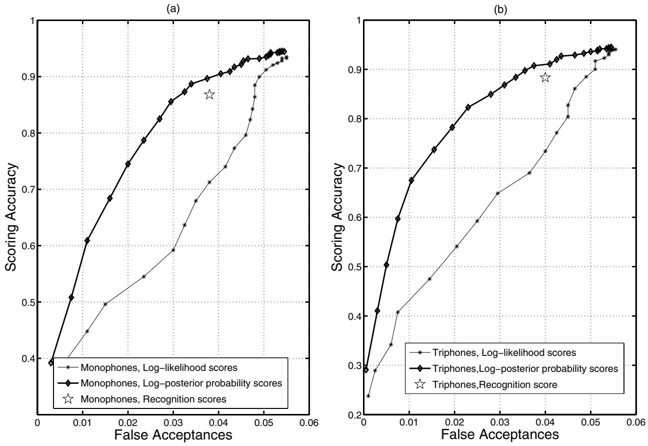


Fig. 7. Scoring accuracy versus false acceptance for different GOP measures

6 Conclusions

In this paper we present our study on automatic detection of tone mispronunciations in Mandarin. After the subjctive evaluations of tone mispronunciations occurred in PSC, two approaches to automatic detection of mispronunciations are presented: tempalted-based and HMM-based. Templates based on 5-level schematic characterization of a tone or the relative comparison with a pitch contour, are proved to be impractical. Statistical MSD-HMM are shown to be more effective and flexible than the template-based approach. Under the HMM framework, different experiments on feature combinations, model types and GOP measures have been compared in terms of recognition and the mispronunciation detections. We observed that MFCC is not as effective for mispronunciation detection as for recognition and the normalization of fundamental frequency in a segment is more useful. Among various GOP measures, log-posterior probability shows the best performance.

References

1. Chen J.-C., Jang J.-S. R., Li J.-Y. and Wu M.-C.: Automatic Pronunciation Assessment for Mandarin Chinese. in Proc. ICME, pp.1979-1982, 2004
2. Wei S., Liu Q.S., Hu Y., Wang R.H.: Automatic Pronunciation Assessment for Mandarin Chinese with Accent, NCMMSC8, pp. 22-25, 2005 (In Chinese)
3. Dong B., Zhao Q.W., Yan Y.H.: Analysis of Methods for Automatic Pronunciation Assessment, NCMMSC8, pp.26-30, 2005, (In Chinese)
4. Franco H., Neumeyer L., Digalakis V., and Ronen O.: Combination of machine scores for automatic grading of pronunciation quality, Speech Communication, vol. 30, pp. 121–130, 2000

5. Witt S.M., and Young S.J.: Computer-assisted pronunciation teaching based on automatic speech recognition. In Language Teaching and Language Technology Groningen, The Netherlands, April 1997
6. Neumeyer L., Franco H., Digalakis V. and Weintraub M.: Automatic Scoring of Pronunciation Quality, Speech Communication, 30: 83-93, 2000
7. Ronen O., Neumeyer L. and Franco H.: Automatic Detection of Mispronunciation for Language Instruction, Proc. European Conf. on Speech Commun. and Technology, pp.645-648, Rodhes, 1997
8. Menzel W., Herron D., Bonaventura P., Morton R.: Automatic detection and correction of non-native English pronunciations, in Proc. of InSTIL, Scotland, pp.49-56, 2000
9. Witt S.M. and Young S.J.: Performance measures for phone–level pronunciation teaching in CALL. in Proc. Speech Technology in Language Learning 1998, Marholmen, Sweden, May 1998
10. Huang C., Chang E., Zhou J.-L., and Lee K.-F.: Accent Modeling Based on Pronunciation Dictionary Adaptation for Large Vocabulary Mandarin Speech Recognition, in Proc. ICSLP 2000, Volume III, pp. 818-821. Oct., 2000
11. Chang E., Zhou J.-L., Di S., Huang C., and Lee,K.-F.: Large Vocabulary Mandarin Speech Recognition with Different Approach in Modeling Tones, in Proc. ICSLP 2000
12. Hirst D. and Espesser R.: Automatic Modelling of Fundamental Frequency Using a Quadratic Spline Function, Travaux de l'Institut de Phontique d'Aixen -Provence, 15, pp.75-85, 1993
13. Tokuda K., Masuko T., Miyazaki N., and Kobayashi T.: Multi-space Probability Distribution HMM, IEICE Trans.Inf. &Syst., E85-D(3):455-464, 2002
14. Wang H.L., Qian Y., Soong F.K.: A Multi-Space Distribution (MSD) Approach To Speech Recognition of Tonal Languages, Accepted by ICSLP 2006
15. Zhou J.-L., Tian Y., Shi Y., Huang C., Chang E.: Tone Articulation Modeling for Mandarin spontaneous Speech recognition, in Proc. ICASSP 2004,pp.997-1000

Towards Automatic Tone Correction in Non-native Mandarin

Mitchell Peabody and Stephanie Seneff

Computer Science and Artificial Intelligence Laboratory,
MIT, Cambridge, MA 02139, USA
{mizhi, seneff}@csail.mit.edu*

Abstract. Feedback is an important part of foreign language learning and *Computer Aided Language Learning* (CALL) systems. For pronunciation tutoring, one method to provide feedback is to provide examples of correct speech for the student to imitate. However, this may be frustrating if a student is unable to completely match the example speech. This research advances towards providing feedback using a student's own voice. Using the case of an American learning Mandarin Chinese, the differences between native and non-native pronunciations of Mandarin tone are highlighted, and a method for correcting tone errors is presented, which uses pitch transformation techniques to alter student tone productions while maintaining other voice characteristics.

1 Introduction

Feedback is essential in foreign language learning, and can take many forms, depending on the particular aspect of speech production being taught. A simple conversation involving teacher feedback is illustrated here:

1 Student “ni3 hao3! wo3 **jiu3** mi2 zhi4.”
2 *Teacher* “*bu2 shi4* ***jiu3****. shi4* ***jiao4****.*”
3 Student “ni3 hao3! wo3 **jiao3** mi2 zhi4.”
4 *Teacher* “*wo3* ***jiao4*** *mi2 zhi4.*”
5 Student “ni3 hao3! wo3 **jiao1** mi2 zhi4.”
6 *Teacher* “***jiao4****. si4 sheng1.*”
7 Student “**jiao4**.”
8 *Teacher* “*hao3.*”
9 Student “ni3 hao3! wo3 **jiao1** mi2 zhi4.”

In this simple conversation, the student, whose primary language is American English, is attempting to say, “Hello! My name is Mitch,” using Mandarin Chinese. The word pronounced incorrectly is presented in bold. In this case, there are two aspects of the pronunciation that are incorrect: the phonetic aspect and the tone aspect.

* This research was funded by the Industrial Technology Research Laboratory and by the Cambridge MIT Institute.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 602–613, 2006.

In order to correct the student's pronunciation, the teacher provides a correct template, in the form of their voice, for the student to imitate. The first correction is to change "jiu" which has the wrong segmental form to "jiao." The second correction is to change the tone of the word from tone 3 to tone 4. The student is immediately able to correct the segmental part of his speech (changing "jiu" to "jiao"), however has trouble correcting the tonal aspect of his speech. In this example, the student unsuccessfully tries to correct his tone within a sentence. He then produces the tone correctly in isolation, but immediately fails to incorporate this change in a sentential context.

One problem is that the student does not know how his voice should sound and only has one reference to base his pronunciation on. The student's anxiety about correctly producing the language may be increased if he is unable to imitate the teacher's voice or identify what the teacher feels is lacking in his pronunciation [1, 2]. Because Mandarin is a tonal language, and the student's primary language, English, is not, the student may have trouble perceiving the distinctions between the tones [3, 4].

A major problem not illustrated by the example is that a teacher is not always available to give the student practice and guidance. However, a *Computer Aided Language Learning* (CALL) system can support practice at any time, in a non-threatening environment, and can provide feedback when a teacher is unavailable. CALL systems, which are designed to facilitate learning a foreign language using a computer, have three essential elements: speaking practice, hearing and understanding practice, and feedback.

A number of methods can be employed to provide the student with correct examples of speech. One method, largely employed by pronunciation dictionaries, is to pre-record native versions of speech being corrected. While this provides very high quality samples of speech for the student to emulate, it suffers from two major flaws. First, it is not scalable in that it is impossible to predict the full range of sentences that could be corrected. Second, the same problem that exists with the teacher is present: a student may still be unable to perceive and correct problems with their own speech. An alternative is to provide samples of speech using a speech synthesizer. This eliminates the scalability problem, but retains the problem of the student's inadequate perception of the error source. It also introduces the additional challenge of providing very high quality speech synthesis, which is very difficult.

If a step back is taken and the target language is considered, another method presents itself. Mandarin is a tonal language which means that tone quality and phonetic quality can be considered independently. Focusing on only the tonal aspect of Mandarin, we propose a method that modifies the *tonally* incorrect portions of student speech to sound correct. We wish to do this in a contextually sensitive manner for entire sentences by predicting an overall pitch contour based on native models. The advantages of this method are that a large database of pre-recorded speech is not required, a speech synthesizer is not utilized, the number of phrases that can be corrected is virtually unlimited, and the feedback is in the student's voice. Furthermore, by listening to two minimally different versions of their spoken utterance, students can tune in to the perceived differences, which pertain directly to tonal aspects.

In this research, a number of questions need to be addressed. What are some characteristics of natively produced tones? How are these different from those produced by non-native speakers? What, if any, variations occur with respect to sentence position? How can the corrections be realized? How can the tone of the speech be modified such that the result has few artifacts? How can the quality of the tones that are produced be tested?

Section 2 provides a brief overview of modern CALL systems that utilize dialogue interaction to allow for student practice. Section 3 gives a brief overview of native Mandarin tones and discusses differences between native and non-native productions of tones. Section 4 discusses our approach to correcting non-native tone errors. Section 5 presents some results and Section 6 summarizes the main points of this paper and provides directions for future work.

2 Background

A common approach to learning in a foreign language classroom is the task-based method. Task-based language learning is a communicative approach in which the student participates in a dialogue with another student, teacher, or native speaker on a particular topic with a specific end goal in mind [5, 6].

Feedback pertaining to various aspects of language learning is given either during or after the conversation. The idea is that, by encouraging the student to come up with sentences and phrases on their own, even if they are imperfect, learning will take place. Feedback may be given to correct major problems, but other problems are allowed to slide.

In recent years, CALL researchers have attempted to enable this form of foreign language learning using dialogue systems. In contrast to tapes or CDs, a computer system has the ability to dynamically create dialogues based on a given scenario for a student. By highly restricting the domain of a lesson to those one might find in a language book, it is feasible to construct dialogues that are dynamic in content and flow.

The *Tactical Language Tutoring System* (TLTS) [7, 8] immerses a student in a 3D world using the *Unreal Tournament 2003* [9] game engine, where he is instructed to accomplish missions by interacting with characters in the environment using speech and non-verbal communication. Speech recognition is done on highly restricted sets of sentences using the Cambridge *Hidden Markov Model Toolkit* (HTK) [10] augmented with noisy-channel models to capture mispronunciations associated with English speakers learning Arabic [11].

Raux and Eskenazi [12] adapted a spoken dialogue system [13] to handle non-native speech through adaptation techniques [14] using a generic task-based dialogue manager [15]. Another key feature of the system is the use of clarification statements to provide implicit feedback through emphasis of certain parts of a student's utterance [16] allowing feedback to be given as part of the dialogue.

Our general approach to CALL [17] is also modeled on the task-based approach. For all of our experiments, we have focused on the Mandarin/English language pair. A prototype system created by Lau [18] was able to carry on short conversations with a student about simple topics such as family relationships. LanguageLand [19] was developed as a multi-modal system intended to help students learn to give directions in a foreign language.

It is within the task-based learning pedagogical framework that we wish to provide feedback. An overview of general pronunciation feedback in CALL can be found in [20]. Our focus here is on feedback as it pertains to pitch, for which a number of strategies have been previously attempted. An oscilloscope system [21] from the early 1960s provided direct visual feedback to the student through a real-time pitch display. A more recent example comes from [22], where a student received feedback in the form of a video game. A simple car driving game indicated to the student the quality of their feedback by how well the car remained in the center of a twisting and curving road.

Instead of explicitly indicating problems with pitch, which only hint at ways to correct the errors, some methods of feedback involve presenting the student with a corrected version of their own voice. For example, the *WinPitchLTL* [23] program provides visual feedback to the student in the form of pitch contours that can be compared against teacher provided models. The program has the additional capability of transforming the pitch of the student's speech to train on aspects such as intonation, stress, or tone. This functionality is obtained through a manual editing process. An automatic method was introduced in [24] where the prosody of isolated words was repaired using *Pitch Synchronous OverLap and Add* (PSOLA) [25–27]. Reference pronunciations were provided by recorded teacher utterances or by KTH's text-to-speech system [28]. Experiments in [29] generated a pitch contour for phrases using linear regression and ToBI [30] transcriptions. The generated contour was compared against a reference contour to show improvement. A similar technique was used in [31] where the authors attempted to repair intonation structure with a focus on stress patterns.

We propose repairing non-native tonal errors in a sentence by producing a model pitch contour based on native data. We examine Chinese tones to determine properties that can be incorporated into this model contour. We also examine differences in tone production between native and non-native speakers.

3 Tone Studies

In this section, we investigate speech data from three corpora: the Yinhe [32] corpus, the Lexical Tone (LT) [33] corpus, and the *Defense Language Institute* (DLI) corpus. The Yinhe data consists of 5,211 Mandarin utterances spoken by native speakers interacting with a dialogue system that provides information about flights and weather. The LT data consists of 497 Mandarin utterances, also in the weather domain, spoken by Americans in their first or second year of studying Mandarin at a college level. The DLI data consists of 5,213 utterances taken from oral proficiency interviews at DLI.

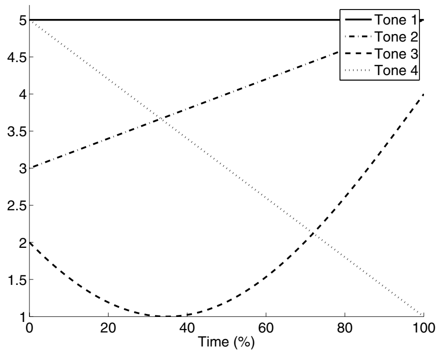


Fig. 1. Canonical forms of f_0 for tones produced in isolated syllables

A tonal language uses pitch, the perception of fundamental frequency (f_0), to lexically distinguish tones. Mandarin Chinese is a tonal language in which every syllable is marked with a tone. Syllables in Chinese are composed of two parts: an initial and a final. The initial phone is either a consonant or the null initial (silence). The final portion of the syllable is composed of vowels and possibly a post-vocalic nasal. The final also functions as the tone bearing unit of a Chinese syllable [34]: the portion of the syllable where pitch differentiates tones.

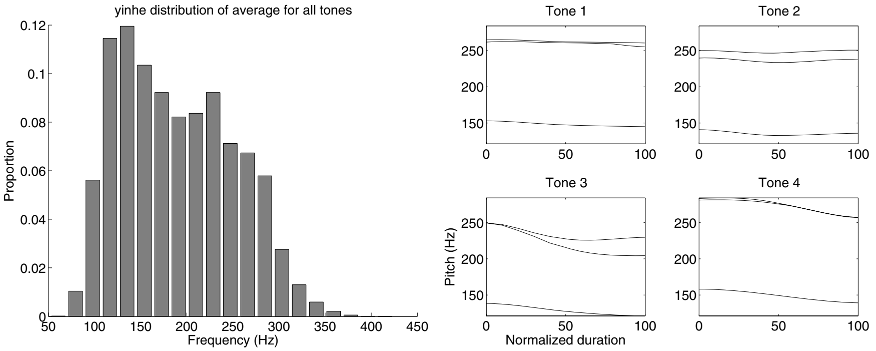
Mandarin has 5 official tones, of which the first four are the most important for understanding. The fifth is often referred to as the neutral tone. Tonal languages lexically distinguish tones using pitch, or f_0 perception, in two main ways: by shape or by absolute height (register). Mandarin tones are mainly distinguished by shape, though there are other perceptual cues [35].

When pronounced in isolation, tones 1 through 4 have shapes that ideally look like those seen in Fig. 1 (tone 5 has no canonical shape, and is not shown). When pronounced as part of a word, phrase, or sentence, the pitch of the tones is altered in complex ways that depend on such factors as left and right contexts, anticipation [36], pitch declination [37], or tone sandhi rules [38].

In general, speakers of a non-tonal language who are learning Mandarin as a foreign language have difficulty both perceiving and producing tone (see, for example [4]). The major preceptual cue for distinguishing Mandarin tones is pitch shape, which makes it a natural starting point for comparison between native and non-native speakers. In order to make meaningful comparisons of shape, the f_0 of the data must be normalized.

Fig. 2a is a histogram of the average f_0 for all voiced portions of speech over the entire Yinhe corpora. The bimodal distribution is due to gender differences in average f_0. Fig. 2b shows the f_0 contours for three randomly selected speakers from the Yinhe corpus: two female and one male. The obvious difference in the average f_0 of the male and females is one of the main reasons for normalization.

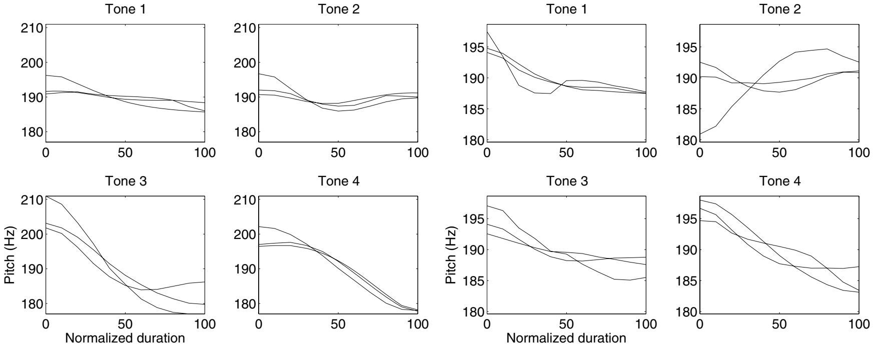
The normalization process has three main steps. First, an overall f_0 value is obtained for the entire Yinhe corpus. For each utterance in both the Yinhe and LT corpora, the f_0 values of each syllable are adjusted to be close to the utterance mean f_0. This step effectively removes tilt due to f_0 declination. Finally, each



(a) Global f_0 distribution.

(b) Raw contours.

Fig. 2. Illustrating the need for normalization of f_0 contours



(a) Native.

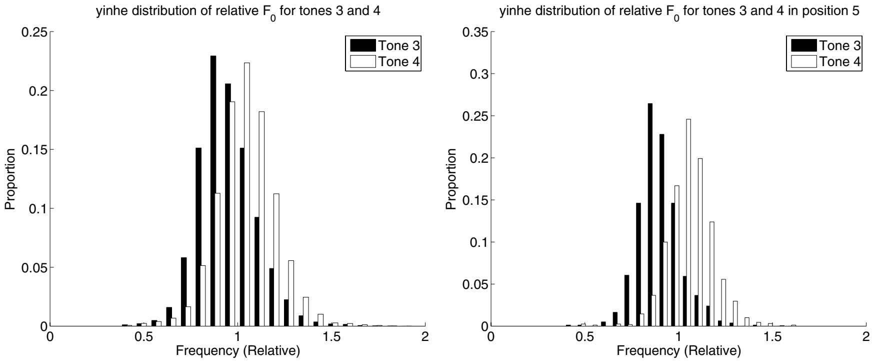
(b) Non-native.

Fig. 3. Comparison of native vs non-native normalized f_0 contours

f_0 in the utterance is scaled by a constant factor to make the utterance mean f_0 equal to the corpus mean f_0 (189.75 Hz). This moves the mean utterance f_0 for each utterance to the same f_0 location as the overall mean for the Yinhe data.

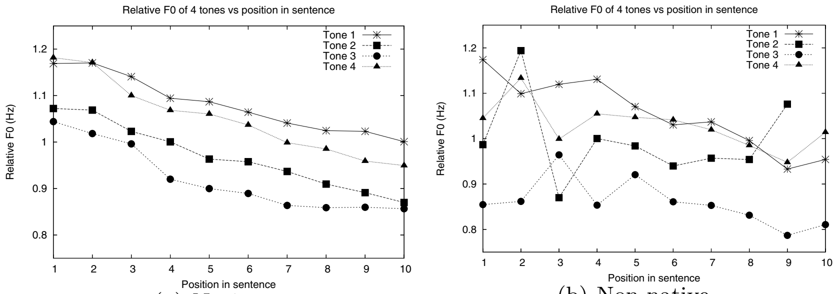
After normalization, native and non-native pitch contours can be compared directly. Fig. 3a shows the f_0 contours for the same three speakers from the Yinhe corpus after normalization. The most important aspect to note is the consistency of the shapes for the native speakers, when contrasted with contours from three random non-native speakers from the LT corpus shown in Fig. 3b. It is evident that these non-native speakers have difficulty producing the contours correctly. For instance, there is very little contrast between the shapes of tone 1 and tone 3 for the non-native speakers.

It is also apparent from analysis of the Yinhe data that Mandarin tones differ in their mean f_0 values, although sentence declination effects must be accounted for to effectively exploit this feature. Averaged over all speakers, tones 1, 2, 3 and 4 have mean f_0 values of 203.73 Hz, 179.37 Hz, 178.05 Hz, and 196.3 Hz



(a) Average over all positions. (b) At syllable position five.

Fig. 4. Separation of tones 3 and 4 by relative f_0



(a) Native. (b) Non-native.

Fig. 5. Relative f_0 declination separated by tone

respectively. Thus, tones 1 and 4 have relatively greater mean values, and tones 2 and 3 have relatively smaller mean values.

Intonation, which encodes phrase level and sentential structure, interacts with tone production in complex ways that are not fully understood. It is known, however, that Mandarin pitch has a generally negative downward slope throughout phrases. This effect, known as pitch declination, must be accounted for explicitly in order to effectively exploit the intrinsic mean f_0 property of tones.

Quantitatively, we can define relative f_0 to be the ratio of f_0, averaged over the duration of the syllable final, over the mean f_0 of the sentence. Fig. 4a shows two histograms illustrating the distribution of relative f_0 for tones 3 and 4, for data over all syllable positions in the sentence. Fig. 4b shows the same plots, but restricted to syllable position 5. It is evident that the two distributions are much better separated when the data are restricted to a single syllable position.

If the relative f_0 of each tone is plotted as a function of syllable position, an interesting picture emerges. Fig. 5a plots the relative f_0 of native Mandarin speakers vs syllable position, and clearly shows that the separation between tones by relative f_0 persists throughout the duration of an utterance. This means that

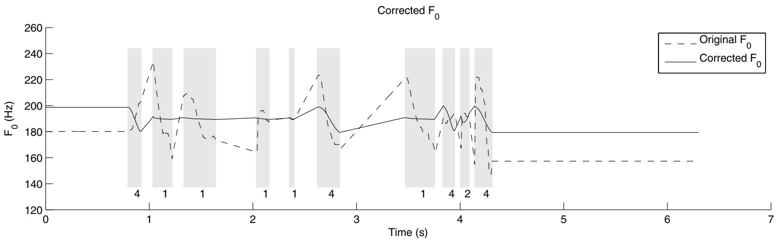


Fig. 6. Corrected contour for the sentence "luo4 shan1 ji1 xing1 qi1 si4 feng1 da4 bu2 da4" (*English: "Will it be windy in Los Angeles on Thursday?"*)

a pitch generating algorithm needs to adjust f_0 for declination based on both syllable position and tone assignment. Intonation generally plays a large role in the quality of pronunciation [39]. As with tone shape, non-native speakers have poor control over the interplay of tone relative f_0 and intonation, as illustrated by Fig. 5b.

4 Approach

Our general approach to providing tonal corrections in sentences is to modify a waveform of the student's speech. This is done in a two stage process. The first stage generates a pitch contour from native tone models. The second stage alters the pitch in the student waveform to match the generated contour.

In the first stage, we assume that an aligned transcription of the correct initials and finals for each syllable in a waveform of student speech is available. The pitch contour is extracted from the original speech using a dynamic programming algorithm described in [40].

For each syllable, the tone assignment for the final portion is determined from the transcription. A series of f_0 values in the shape of the tone is generated over the duration of the final. The f_0 values are adjusted to be appropriate for the current syllable position according to a declination model. For those time segments in which there is no final (and hence, no tone), the f_0 values are linearly interpolated to make the contour continuous.

Tone shapes are represented by four coefficients from the discrete Legendre transform as described in [41]. The model f_0 contour can be reconstructed from the first four coefficients. Parametrically characterizing the pitch contour of the tones has two benefits: pitch contours for different syllable durations can be easily generated, and less training data is required for each tone model.

The intonation declination models are linear equations derived from a regression on the first 10 syllables of the relative f_0 for each tone. For each tone, this gives a parametric model that can be used to adjust the f_0 values for a given tone at a given syllable position.

An example of a corrected utterance can be seen in Fig. 6. Normalization for speaker f_0 range and for sentence declination have been incorporated into the connected contour plot, and thus it has a very flat declination and correctly shaped tones.

In this example, there are very evident changes in the shapes of the tone contours. For example, in syllable position one, the syllable "luo4" is seen. This tone should have a falling pitch, but the speaker produced it with a rising pitch. The generation algorithm has produced a contour that is qualitatively closer to the correct native contour.

In the second stage, the pitch contour of the original speech and the generated pitch contour are both available. The speech in the student waveform is processed so that the pitch at each time interval is adjusted according to the generated version. The adjustments are done using a phase vocoder as described in [42, 43], which allows pitch to be adjusted up or down depending on a real-valued factor.

The advantage of using a phase vocoder is two-fold: it can produce very high quality pitch transformations and it can do these transformations by manipulating the waveform itself; no pitch extraction and resynthesis is required. In previous usage, the phase vocoder had adjusted the pitch of speech by a constant factor, but for this application, the pitch needs to be adjusted by a different factor for each time frame.

The end result of this algorithm is to produce a version of the student waveform that has been corrected for tone. The voice can still be identified as originally belonging to the speaker, but the tones will sound closer to native quality.

Table 1. Classification accuracy for original and generated contours

Dataset	# Utts	Original	Predicted
LT	497	41.3%	92.2%
DLI	5213	29.0%	81.3%

5 Results

To evaluate the quality of the generated tone contours, we used a tone classifier trained on native data to classify tones from both original pitch contours and from the corresponding generated contours. The reasoning is that, if the generated pitch contour is closer to native quality, then the classification accuracy for a given utterance should be much better than for the original pitch contours. The choice of using a classifier to evaluate quality was motivated by the expectation that, in the future, this research will be incorporated into a larger CALL system that has automatic tone evaluation. We wished to establish that any corrective guidance regarding tone would be detectable by such a method.

The native training data used was the Yinhe data with normalized f_0. This normalization corrected for both syllable position and speaker pitch range. The feature vector used to train the classifier models was composed of the four Legendre coefficients that parameterize the tone shapes. Each tone model in the classifier was composed of 16 Gaussian mixture models.

Pitch contours were generated, as described in Section 4, for each utterance in the LT and DLI corpora, which contain non-native data. Table 1 shows the accuracy of the classifier on the original pitch contours and on the generated pitch contours. For both the LT and DLI corpora, there is a large increase in classification accuracy.

6 Summary and Future Work

This paper proposes a novel method for pronunciation feedback for learners of Mandarin by providing students with a corrected version of their own speech. An examination of native and non-native productions of tone revealed that non-native speakers have difficulty producing Mandarin tones. Based on native Mandarin speech, models for tones and phrase declination were built that were used to generate a pitch contour for a given utterance spoken by a non-native speaker. The results indicate that this generated pitch contour produces tones that are much closer to native quality than the original non-native speech.

In order to correct the student's speech we need to reintroduce the sentence declination into the generated pitch contour, and then apply the phase vocoder technique to instantiate it. Implementation of this portion of the algorithm is planned for the immediate future. While the generated contours were found to be much closer to native quality than the original contours; there is not yet any indication that there is correlation with human perception. The utterances produced by the phase vocoder need to be evaluated for improved tone quality through listening tests conducted by native speakers of Mandarin.

The tone models represented the lexical tones averaged over all left and right contexts; however contextual variations should be accounted for explicitly in the models. Modeling these contextual variations into account will also help capture prosodic phenomena such as tone sandhi rules. To do this will require more non-native data, as context specific models will experience data-sparseness issues.

This research dealt explicitly with feedback with the assumption that all tones were produced incorrectly by the non-native speakers. Most likely, though, only some of the tones will be produced incorrectly. In the near future, data will be marked by a fluent Mandarin speaker for tone quality. Based on feature comparisons between native and non-native speakers, methods for detecting which tones are produced incorrectly will be explored. This will allow for more selective feedback to be given.

Acknowledgement

We would like to thank Chao Wang for providing us with the pitch detector as well as significant technical assistance for this research.

References

1. Horwitz, E.K., Horwitz, M.B., Cope, J.: Foreign language classroom anxiety. The Modern Language Journal **70**(2) (1986) 152–132
2. Onwuegbuzie, A.J., Bailey, P., Daley, C.E.: Factors associated with foreign language anxiety. Applied Psycholinguistics **20** (1999) 217–239

3. Kiriloff, C.: On the auditory discrimination of tones in mandarin. Phonetica **20** (1969) 63–67
4. Leather, J.: Perceptual and productive learning of chinese lexical tone by dutch and english speakers. In Leather, J., James, A., eds.: New Sounds 90, University of Amsterdam (1990) 72–97
5. Skehan, P.: Task-based instruction. Language Teaching **36**(01) (2003) 1–14
6. Ellis, R.: Task-based language learning and teaching. Oxford University Press, Oxford, UK (2003)
7. Johnson, L., Marsella, S., Mote, N., Viljhálmsson, H., Narayanan, S., Choi, S.: Tactical language training system: Supporting the rapid acquisition of foreign language and cultural skills. In: Proc. of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems. (2004)
8. Johnson, L., Beal, C.R., Fowles-Winkler, A., Lauper, U., Marsella, S., Narayanan, S., Papachristou, D., Vilhjálmsson, H.: Tactical language training system: An interim report. In: Intelligent Tutoring Systems. (2004) 336–345
9. Epic Games, I.: Unreal tournament 2003. http://www.unrealtournament.com/ (2003)
10. Young, S., Odell, J., Ollason, D., Valtchev, V., Woodland, P.: The HTK Book. Cambridge University, Cambridge, UK (1997)
11. Mote, N., Johnson, L., Sethy, A., Silva, J., Narayanan, S.: Tactical language detection and modeling of learner speech errors: The case of arabic tactical language training for american english speakers. In: Proc. of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems. (2004)
12. Raux, A., Eskenazi, M.: Using task-oriented spoken dialogue systems for language learning: Potential, practical applications and challenges. In: Proc. of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems. (2004)
13. Raux, A., Langner, B., Eskenazi, M., Black, A.: Let's go: Improving spoken dialog systems for the elderly and non-natives. In: Eurospeech '03, Geneva, Switzerland (2003)
14. Raux, A., Eskenazi, M.: Non-native users in the let's go!! spoken dialogue system: Dealing with linguistic mismatch. In: HLT/NAACL 2004, Boston, MA (2004)
15. Bohus, D., Rudnicky, A.: Ravenclaw: Dialog management using hierarchical task decomposition and an expectation agenda. In: Eurospeech '03, Geneva, Switzerland (2003)
16. Raux, A., Black, A.: A unit selection approach to $f0$ modeling and its application to emphasis. In: ASRU 2003, St Thomas, US Virgin Islands (2003)
17. Seneff, S., Wang, C., Peabody, M., Zue, V.: Second language acquisition through human computer dialogue. In: Proceedings of ISCSLP. (2004)
18. Lau, T.L.J.: Slls: An online conversational spoken language learning system. Master's thesis, Massachusetts Institute of Technology (2003)
19. Lee, V.: Languageland: A multimodal conversational spoken language learning system. Master's thesis, Massachusetts Institute of Technology (2004) MEng.
20. Neri, A., Cucchiarini, C., Strik, H.: Feedback in computer assisted pronunciation training: technology push or demand pull? In: Proceedings of ICSLP, Denver, USA (2002) 1209–1212
21. Vardanian, R.M.: Teaching english through oscilloscope displays. Languate Learning **3**(4) (1964) 109–118
22. Álvarez, A., Martínez, R., Gómez, P., Domínguez, J.L.: A signal processing technique for speech visualization. In: STILL, ESCA, ESCA and Department of Speech, Music and Hearing KTH (1998)

23. Martin, P.: Winpitch ltl ii, a multimodel pronunciation software. In: Proc. of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems. (2004)
24. Sundström, A.: Automatic prosody modification as a means for foreign language pronunciation training. In: STILL, ESCA, ESCA and Department of Speech, Music and Hearing KTH (1998) 49–52
25. Hamon, C., Moulines, E., Charpentier, F.: A diphone synthesis system based on time-domain prosodic modifications of speech. In: Proc. ICASSP '89, Glasgow, Scotland (1989) 238–241
26. Moulines, E., Charpentier, F.: Pitch synchronous waveform processing techniques for text-to-speech conversion using diphones. Speech Communication **9** (1990) 453–467
27. Moulines, E., Laroche, J.: Non-parametric techniques for pitch scaling and time-scale modification of speech. Speech Communication **16**(2) (1995) 175–207
28. Carlson, R., Granström, B., Hunnicutt, S.: Multilingual text-to-speech development and applications. In Ainsworth, A., ed.: Advances in speech, hearing and language processing. JAI Press, London (1990) 269–296
29. Black, A.W., Hunt, A.J.: Generating f0 contours from tobi labels using linear regression. In: Proceedings of the Fourth International Conference on Spoken Language Processing. Volume 3. (1996) 1385–1388
30. Silverman, K.E.A., Beckman, M., Pitrelli, J.F., Ostendorf, M., Wightman, C., Price, P., Pierrehumbert, J., Hirschberg, J.: Tobi: A standard for labeling english prosody. In: Proceedings of the 1992 International Conference on Spoken Language Processing. Volume 2., Banff, Canada (1992) 867–870
31. Jilka, M., Möhler, G.: Intonational foreign accent: Speech technology and foreign language testing. In: STILL, ESCA, ESCA and Department of Speech, Music and Hearing KTH (1998) 115–118
32. Wang, C., Glass, J.R., Meng, H., Polifroni, J., Seneff, S., Zue, V.: YINHE: A Mandarin Chinese version of the GALAXY system. In: Proc. EUROSPEECH'97, Rhodes, Greece (1997) 351–354
33. Peabody, M., Seneff, S., Wang, C.: Mandarin tone acquisition through typed interactions. In: Proc. of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems. (2004)
34. Duanmu, S.: The Phonology of Standard Chinese. Oxford University Press (2002)
35. Whalen, D., Xu, Y.: Information for mandarin tones in the amplitude contour and in brief segments. Phonetica **49** (1992) 25–47
36. Xu, Y.: Contextual tonal variations in mandarin. Journal of Phonetics **25** (1997) 61–83
37. Shih, C.: Declination in mandarin. Prosody tutorial at 7th International Conference on Spoken Language Processiong (2002)
38. Chen, M.: Tone Sandhi: Patterns Across Chinese Dialects. Cambridge University Press, Cambridge, UK (2000)
39. Jilka, M.: The contribution of intonation to the perception of foreign accent. PhD thesis, University of Stuttgart (2000)
40. Wang, C., Seneff, S.: Robust pitch tracking for prosodic modeling in telephone speech. In: Proc. ICASSP, Istanbul, Turkey (2000) 887–890
41. Wang, C.: Prosodic Modeling for Improved Speech Recognition and Understanding. PhD thesis, Massachusetts Institute of Technology (2001)
42. Seneff, S.: System to independently modify excitation and/or spectrum of speech waveform without explicit pitch extraction. IEEE Trans. Acoustics, Speech and Signal Processing **ASSP-30**(4) (1982) 566
43. Tang, M., Wang, C., Seneff, S.: Voice transformations: From speech synthesis to mammalian vocalizations. In: Proc. Eurospeech 2001, Aalborg, Denmark (2001)

A Corpus-Based Approach for Cooperative Response Generation in a Dialog System

Zhiyong Wu, Helen Meng, Hui Ning, and Sam C. Tse

Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong, Shatin
john.zy.wu@gmail.com, {hmmeng, hning, cftse}@se.cuhk.edu.hk

Abstract. This paper presents a corpus-based approach for cooperative response generation in a spoken dialog system for the Hong Kong tourism domain. A corpus with 3874 requests and responses is collected using Wizard-of-Oz framework. The corpus then undergoes a regularization process that simplifies the interactions to ease subsequent modeling. A semi-automatic process is developed to annotate each utterance in the dialog turns in terms of their key concepts (KC), task goal (TG) and dialog acts (DA). TG and DA characterize the informational goal and communicative goal of the utterance respectively. The annotation procedure is integrated with a dialog modeling heuristic and a discourse inheritance strategy to generate a semantic abstraction (SA), in the form of {*TG*, *DA*, *KC*}, for each user request and system response in the dialog. Semantic transitions, i.e. $\{TG, DA, KC\}_{\text{user}} \rightarrow \{TG, DA, KC\}_{\text{system}}$, may hence be directly derived from the corpus as rules for *response message planning*. Related verbalization methods may also be derived from the corpus and used as templates for *response message realization*. All the rules and templates are stored externally in a human-readable text file which brings the advantage of easy extensibility of the system. Evaluation of this corpus based approach shows that 83% of the generated responses are coherent with the user's request and qualitative rating achieves a score of 4.0 on a five-point Likert scale.

Keywords: Natural language generation (NLG), Response generation, Corpus-based approach.

1 Introduction

Continual advancements in speech and language technologies have brought usable spoken dialog systems (SDS) within reach. SDS typically supports goal-oriented human-computer conversations regarding restricted application domains, e.g. asking for a restaurant recommendation, planning a trip, etc. SDS integrates technologies including speech recognition (SR), natural language understanding (NLU), dialog modeling, information/database access and text-to-speech synthesis. An indispensable component that facilitates effective two-way, human-computer interaction is natural language generation (NLG) of *cooperative system responses* that tailor to the user's information needs and linguistic preferences. NLG is defined as the process of transforming a semantic specification from the dialog model (DM) into a *semantically*

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 614 – 626, 2006.

well-posed and *syntactically well-formed* message. The message can be presented to the user as on-screen text and/or synthesized speech. The demarcation between the DM and NLG may vary from one system to another. Some (earlier) systems do not distinguish between the two processes. In this work, the demarcation is drawn whereby the DM provides discourse-inherited semantics for the NLG. The NLG aims to compose a well-posed and well-formed message that can serve as a cooperative system response. To compose a well-posed message, the NLG needs to select content pertinent to the current dialog turn and cast the content in a message plan that is coherent and succinct. To compose a well-formed message, the NLG needs to select syntactic and elements for textual/audio realization of the response message. We divide NLG problem into two sub-problems – (i) *message planning* formulates a well-posed message plan (MP) based on relevant semantics; and (ii) *message realization* generates a well-formed linguistic realization from the MP.

Previous approaches in NLG generally fall within a continuum between the non-linguistic template-based approach and the fully linguistic approach [1,2]. The template-based approach has been widely adopted due to ease of development, maintenance and predictability [3]. However, handcrafting templates for every application domain is a tedious process with low portability. It is often impossible to handcraft templates that fully cover the combinatoric space of communicative goals and discourse contexts. Hence templates offer limited variety and the approach becomes untenable as the application domain grows. Fully linguistic approaches mostly originate from research in NLG of monologs (e.g. reports, summaries, etc.) and incorporate a huge amount of linguistic knowledge [4,5]. Adapting these approaches for dialogs in restricted domains and achieving real-time performance may be difficult [6]. Recent efforts in NLG research strive to strike a balance between the non-linguistic and fully linguistic ends of the spectrum, by using simple rules/grammars augmented with corpus-based statistics. This can reduce the need for a full linguistic characterization and can also introduce variety into the NLG output [7]. A representative example is the use of stochastically combined dialog acts to form surface realizations and these are then selected by a filter trained on a human-graded corpus [8].

In this paper, we present an approach where message planning strategies and message realization templates are derived from a dialog corpus. This approach can vastly reduce the human effort that needs to be devoted to authoring rules and templates. We collected a dialog corpus by means of a Wizard-of-Oz setup, where the "wizard" attempts with best effort to answer to the user's inquiries in a systematic and succinct way. The collected data then undergoes a manual "regularization" process for simplification in order to ease subsequent modeling. We also designed a semantic abstraction of each user's request and system's response, in terms of key concepts, tasks goals (i.e. the informational goals) and dialog acts (i.e. communicative goals). Hence we may capture the message planning strategies found in the corpus through semantic transitions of a pair of request/response turns. For a given message plan, we may also refer to the corpus to derive message realization templates. This corpus-based approach eases development of the NLG component and may enhance portability across languages and applications. In the following, we present the details of corpus development, semantic abstraction and annotation, message planning, message realization as well as evaluation results.

2 Corpus Development

2.1 Information Domain

The information domain is specific to Hong Kong tourism, as defined by the Tourism Board's website – Discover Hong Kong.[1] The domain covers information ranging from scenic attractions, shopping attractions, transportation, fare prices, events, tours, etc. This diversity is useful for our current research in natural language generation.

Based on the website, we also developed a database covering 349 attractions. Related information constituents that are tagged with XML (eXtensible Markup Language) include name, type, description, routing, time, url, etc. An example is shown in Table 1 for illustration.

Table 1. An example of XML-tagged data entry in the Hong Kong tourism domain

```
<ATTRACTION>
  <NAME>迪士尼樂園</NAME>   (translation: Disneyland)
  <TYPE>主題公園</TYPE>   (theme park)
  <DESCRIPTION>
  從踏進香港迪士尼樂園那一刻開始，令人興奮着迷的奇妙之旅便已展開！……
  </DESCRIPTION>   (the Hong Kong Disneyland is an exciting place...)
  <ROUTE>在地鐵欣澳站轉乘迪士尼綫列車</ROUTE>   (take the mass transit railway
  (MTR) to Sunny Bay station and transit to the Disney line)
  <TIME>開放時間為上午10時至晚上8時</TIME>   (opening hours...)
  <PRICE>成人295，小童210，長者170</PRICE>   (fares for adults, children and seniors)
  <URL>http://hongkongdisneyland.com</URL>
</ATTRACTION>
```

2.2 Eliciting Interactions Using a Wizard-of-Oz Data Collection

In order to elicit interactions in the selected domain, we use a Wizard-of-Oz (WoZ) data collection setup to elicit interactions from a group of thirty invited subjects. Each subject and the wizard sat in different rooms, and interacted through a multimodal and multimedia interface through networked computers. The subjects can issue inquiries using speech, typed text and/or pen gestures. The wizard can refer to the Discover Hong Kong website during the entire data collection process and always tries to respond to the user's inquiries with best effort. All interactions were logged by the system. As a result of this data collection process, we have a series of dialogs that contain rather free-form wizard-generated responses for the subjects' inquiries. These enable us to define the major informational goals (or task goals, TG) of the subjects, dialog acts (DA) that characterize the course of dialog interactions and related key concepts (KC) that may come from the current interaction (dialog turn) or inherited from previous interactions.

[1] http://www.discoverhongkong.com

2.3 Data Regularization Process

As mentioned earlier, the wizard's responses as logged from the WoZ data collection procedure is relatively free form. It contains many disfluencies such as filled pause, word order reversal due to spontaneity in interaction and tagged information indicating responses in alternative modalities, e.g. highlighted points on a map, urls, etc. In order to ease the subsequent process of modeling the dialog responses, we devised a manual procedure of data regularization where the collected data are simplified into short sentences/utterances with straightforward structures. This paves the way of easing the development of message templates for verbalization of relevant information content. In total, we have regularized the entire dialog corpus, which consists of 1500 dialog turns, each with two to five utterances. Overall, there are 3874 request and response utterances. Table 2 shows a simple dialog interaction before and after the data regularization process.

Table 2. An example dialog between the Wizard (W) and the User (U) before (Original data) and after (Refined data) the data regularization process

	Original data	Refined data
W0	請問，你第二天想去哪裏？	請問，你第二天想去哪裏？ *(Where would you like to go on the second day?)*
U1	Er，主題公園…我想去看看。	我想去主題公園看看。*(I would like to visit theme park)*
W1	請問你想去海洋公園，還是迪士尼樂園？	請問你想去海洋公園，還是迪士尼樂園？ *(Would you like to visit Ocean Park or Disneyland?)*
U2	Um，讓我想想。去迪士尼樂園好了。	讓我想想。去迪士尼樂園好了。 *(Let me think. I prefer Disneyland.)*
W2	這裏是迪士尼樂園的資料<url>，請看。	這裏是迪士尼樂園的資料，請看。*(Here is the information about Disneyland, please have a look.)*
U3	從中環到這裏<point: pictures>怎麼走呢？	從中環到這裏怎麼走呢？ *(How could I get here from Central?)*
W3	Er，從中環到迪士尼樂園的話，你可以在欣澳站轉乘地鐵迪士尼綫列車就到了。	從中環到迪士尼樂園，你可以在地鐵欣澳站轉乘迪士尼綫列車。*(From Central to Disneyland, you can take the MTR to Sunny Bay station and transit to the Disney line.)*
U4	那麼，有沒有那個Er海洋公園的資料？	有沒有海洋公園的資料？ *(Is there any introduction about Ocean Park?)*
W4	這個就是海洋公園的資料<url>，請看。	這個就是海洋公園的資料，請看。*(This is the information about Ocean Park, please have a look.)*
U5	再見。	再見。*(Bye-bye.)*
W5	祝你旅途愉快！	祝你旅途愉快！ *(Have a good trip!)*

3 Semi-automatic Corpus Annotation of Semantic Constituents

A critical stage in corpus development is the annotation of major semantic constituents in the collected data. These semantic constituents must characterize: (i) what are the types of questions asked; (ii) what kinds of content are necessary for answering these questions (i.e. *response message planning*); and (iii) how such content should be expressed (i.e. *response message realization*). As mentioned above, we believe that the major semantic constituents needed include the key concepts (KC) in a verbal message; the domain-specific task goal (TG) underlying the message; as well as the communicative role of the message in the course of the dialog, as symbolized by the dialog act (DA) [9,10]. We have devised a semi-automatic method of annotating such semantic constituents. The objective is to reduce the manual effort needed, speed up the annotation process, as well as to enhance consistency in the annotations.

3.1 Tagging Key Concepts (KC)

We defined approximate 800 grammar rules (in the form of regular expressions) from analyzing the collected data for tagging concepts. Examples are shown in Table 3.

Table 3. Example of grammar rules for tagging key concepts (KC)

<table>
<tr><td>attraction → 迪士尼樂園 | 海洋公園 | 主題公園 | … <i>(Disneyland | Ocean park | Theme park |..)</i></td><td></td></tr>
<tr><td>how → 怎麼 | 如何 | …</td><td><i>(These Chinese tokens mean “how to”)</i></td></tr>
<tr><td>go → 走 | 行 …</td><td><i>(These Chinese tokens mean “go or walk”)</i></td></tr>
<tr><td>origin → 從 [attraction]</td><td><i>(from [attraction])</i></td></tr>
<tr><td>destination → 到 [attraction]</td><td><i>(to [attraction])</i></td></tr>
<tr><td>directions → [how] [go]</td><td></td></tr>
</table>

3.2 Task Goals and Dialog Acts

The task goal (TG) symbolizes the information goal of the user's request and is domain-specific. The dialog act (DA) expresses the communicative goal of an expression in the course of a dialog and bears relationships with the neighboring dialog turns. The DA is largely domain-independent. We defined 12 TGs based on the collected corpus, as shown in Table 4. We also included 17 DAs, adapted from VERBMOBIL-2 [9], as shown in Table 5.

Table 4. 12 Hong Kong tourism domain specific task goals (TGs)

ATTRACTION, DURATION, FEE, LOCATION, PHONE, ROUTE, SHOPING, HOURS, TOURING, FOOD, HOTEL, RESERVATIONS

Table 5. 17 domain independent dialog acts (DAs)

APOLOGY, BYE, BACKCHANNEL, CLOSE, CONFIRM, DEFER, GREET, SUGGEST, THANK, FEEDBACK_NEGATIVE, FEEDBACK_POSITIVE, REQUEST_SUGGEST, REQUEST_COMMENT, REQUEST_DETAILS, REQUEST_PREFERENCE, INFORM_GENERAL, INFORM_DETAILS

3.3 Semi-automatic Annotation Process

Each dialog turn in the regularized corpus is segmented into individual utterances such that each utterance corresponds to only one TG and one DA. For example, the second user’s request (U2) “讓我想想。去迪士尼樂園好了。*(Let me think. I prefer Disneyland.)*” (see Table 2) is segmented into two utterances as shown in Table 7 – “讓我想想 *(Let me think.)*” followed by “去迪士尼樂園好了 *(I prefer Disneyland.)*”.

Table 6. Division of the Corpus into four data subsets

Data subset	#1	#2	#3	#4
Number of utterances	948	939	986	1001

Table 7. Results of annotation, based on the example presented earlier in Table 2

W0	請問，你第二天想去哪裏？ KC: {ask_where=去哪裏} TG: ATTRACTION	 DA: REQUEST_PREFERENCE
U1	我想去主題公園看看。 KC: {attraction=主題公園} TG: ATTRACTION	 DA: INFORM_DETAILS
W1	請問你想去海洋公園，還是迪士尼樂園？ KC: {attraction=海洋公園, attraction=迪士尼樂園} TG: ATTRACTION	 DA: REQUEST_COMMENT
U2	讓我想想。 KC: {think=想想} TG: ATTRACTION	 DA: DEFER
U2	去迪士尼樂園好了。 KC: {attraction=迪士尼樂園} TG: ATTRACTION	 DA: INFORM_DETAILS
W2	這裏是迪士尼樂園的資料，請看。 KC: {attraction=迪士尼樂園} TG: ATTRACTION	 DA: INFORM_GENERAL
U3	從中環到這裏怎麼走呢？ KC: {origin=中環, destination=這裏, directions=怎麼走} TG: ROUTE	 DA: REQUEST_DETAILS
W3	從中環到迪士尼樂園，你可以在地鐵欣澳站轉乘迪士尼綫列車。 KC: {origin=中環, destination=迪士尼樂園, route=在地鐵..} TG: ROUTE	 DA: INFORM_DETAILS
U4	有沒有海洋公園的資料？ KC: {attraction=海洋公園} TG: ATTRACTION	 DA: REQUEST_DETAILS
W4	這裏就是海洋公園的資料，請看。 KC: {attraction=海洋公園} TG: ATTRACTION	 DA: INFORM_GENERAL
U5	再見。 KC: {bye=再見} TG: ATTRACTION	 DA: BYE
W5	祝你旅途愉快！ KC: {good_trip=旅途愉快} TG: ATTRACTION	 DA: CLOSE

We divided the corpus into four subsets, as shown in Table 6. The annotation procedure is incremental. We first hand-annotate data subset #1 in terms of TG and DA. KCs are tagged by the regular expressions mentioned above. All annotations (KC, TG and DA) are checked by hand and are used to train a suite of Belief Networks (BNs) [11] that can accept a series of input KCs from an utterance and output the TG and DA labels for the utterance. These BNs are used to label data subset #2 which then undergoes a pass of manual checking. Thereafter, both data subsets #1 and #2 are used to retrain the BNs and these are subsequently used to label data subset #3, which again undergoes a pass of manual checking. Thereafter, all three data subsets are used to retrain the BNs and these are evaluated based on data subset #4. The BNs achieve an accuracy of 79% for TG and 77% for DA labeling in data subset #4. Table 7 illustrates an example of the end result of this annotation process. Every utterance in a dialog turn may thus be annotated with KCs, TG and DA.

4 Message Planning and Realization in Response Generation

The annotation procedure described in the previous section is applied to every user's request and system (wizard) response in the regularized corpus. In addition, our dialog model (DM) incorporates a heuristic that the TG of the ensuing response is assumed to be identical to the user's request since the system (wizard) is generating cooperative responses. The DM also incorporates a *selective discourse inheritance strategy* [12] to enhance the completeness of the semantic representation of an utterance. For example, if the user first asks "海洋公園怎麼去？" *(How can I get to Ocean Park?)*, followed by "迪士尼樂園呢？" *(How about Disneyland?)*, the second question must inherit appropriate concepts from the previous question in order to have a self-complete meaning. We have developed a set of context-dependent inheritance rules [12] that govern the inheritance of TG or KC from previous dialog turns. The extensions of the heuristic and discourse inheritance raised the TG and DA label accuracies to over 90% in data subset #4. Sequential processing by the semi-automatic annotation process and the DM transforms every user request and system (wizard) response in the collected corpus into a *succinct semantic representation*, in terms of $\{TG, DA, KC\}$. Such semantic abstraction (SA) of user's request and system's responses are useful for deriving strategies of response message planning as well as methods of response message realization. We will describe the two procedures in the following.

4.1 Strategies for Response Message Planning

Parsing for the TG, DA and KC in a regularized user request or system (wizard) response message automatically generates a semantic abstraction (SA) representation $\{TG, DA, KC\}$. Pairing up the SAs of a user's request with its system response in the subsequent dialog turn automatically derives message planning strategies in the form of semantic transitions, i.e.:

$$\{TG, DA, KC\}_{\text{user}} \rightarrow \{TG, DA, KC\}_{\text{system}}.$$

In other words, each pair of user-system interactions in the 3,874 utterances in our corpus offer one instance of message planning by the wizard in the context of the dialog system. Hence, our strategies for message planning are automatically derived in a data-driven manner.

We analyzed these instances and noted several features:

(i) A user's dialog turn may contain multiple utterances and each has its own SA representation. In such situations, the semantic transition rule is based only on the last utterance and its SA representation. This is because our corpus suggests that the last utterance can fully characterize the user's dialog turn. For example, the second user turn in Table 7 will only derive the SA representation of $\{\mathit{ATTRACTION}, \mathit{INFORM\_DETAILS}, \mathit{attraction}\}_{\text{user}}$.

(ii) It is possible for different $\{TG, DA, KC\}_{\text{user}}$ to transit to the same $\{TG, DA, KC\}_{\text{system}}$. For example, in Table 7 the pair of dialog turns (U2, W2) produces $\{\mathit{ATTRACTION}, \mathit{INFORM\_DETAILS}, \mathit{attraction}\}_{\text{user}}\rightarrow\{\mathit{ATTRACTION}, \mathit{INFORM\_GENERAL}, \mathit{attraction}\}_{\text{system}}$; while the pair of dialog turns (U4, W4) produces $\{\mathit{ATTRACTION}, \mathit{REQUEST\_DETAILS}, \mathit{attraction}\}_{\text{user}}\rightarrow\{\mathit{ATTRACTION}, \mathit{INFORM\_GENERAL}, \mathit{attraction}\}_{\text{system}}$.

(iii) It is also possible for a given $\{TG, DA, KC\}_{\text{user}}$ to transit to several possible $\{TG, DA, KC\}_{\text{system}}$. For example, in Table 7, the pair of dialog turns (U1, W1) produces $\{\mathit{ATTRACTION}, \mathit{INFORM\_DETAILS}, \mathit{attraction}\}_{\text{user}}\rightarrow\{\mathit{ATTRACTION}, \mathit{REQUEST\_COMMENT}, \mathit{attraction}\}_{\text{system}}$. However, the pair of dialog turns (U2, W2) produces $\{\mathit{ATTRACTION}, \mathit{INFORM\_DETAILS}, \mathit{attraction}\}_{\text{user}}\rightarrow\{\mathit{ATTRACTION}, \mathit{INFORM\_GENERAL}, \mathit{attraction}\}_{\text{system}}$. This presents the need for devising a set of *rule selection conditions* in message planning. An illustration is presented in Table 8, where Rules 1 to 4 are all possible transitions originating from the same $\{TG, DA, KC\}_{\text{user}}$. It should be noted that these rule selection conditions are inserted manually upon analysis of the corpus. However, as illustrated in Table 8, these simple conditions should be generalizable to other information domains.

Table 8. An example of semantic transition rules which constitutes the message planning strategies for cooperative response generation. Rule selection conditions may be applied if there are multiple possible message plan options. These conditions may contain key concepts (denoted by '#') whose values are obtained either from database retrieval results (denoted by *database#concept*) or from the parsed user request (denoted by *request#concept*).

Semantic Transition Rule Format
{TG, DA, KC}user→{TG, DA, KC}system

Rule 1
{ATTRACTION, INFORM_DETAILS, attraction}user→
    {ATTRACTION, REQUEST_COMMENT, place}system

Rule 2
{ATTRACTION, INFORM_DETAILS, attraction}user→
    {ATTRACTION, INFORM_GENERAL, attraction}system

Rule 3
{ATTRACTION, INFORM_DETAILS, attraction}user→
    {ATTRACTION, INFORM_DETAILS, attraction}system

Rule 4
{ATTRACTION, INFORM_DETAILS, attraction}user→
    {ATTRACTION, APOLOGY, sorry}system

Control Conditions for the above Rules Selection

```
IF ({database#result_number}>1) THEN select Rule 1
ELSEIF ({database#result_number}==0) THEN select Rule 4
ELSEIF ({request#detail}!=null) THEN select Rule 3
ELSEIF ({database#url}!=null) || ({database#picture}!=null) THEN select Rule 2
```

The introduction of *rule selection conditions* adds context-dependent variability in cooperative response generation. Referring to the dialog in Table 7 and the conditions in Table 8, the various conditions are:

- For the user request U1 in Table 7, the system finds several matching attractions related to the concept "attraction=主題公園 *(theme park)*" in the database. Hence the first rule selection condition {database#result_number}>1 in Table 8 is satisfied and Rule 1 is used as the message plan. Hence the system presents all matching options to the user in the generated response (W1) and seeks the user's input by the dialog act REQUEST_COMMENT.
- The user's feedback in U2 of Table 7 sets the concept value of "attraction=迪士尼樂園 *(Disneyland)*". The system can only find one matching entry in the database which comes with URL information. Hence the fourth rule selection condition in Table 8 {database#url}!=null is satisfied and Rule 2 is used as the message plan. This generates the response W2 under the dialog act of INFORM_GENERAL.
- If the user were to follow up with an utterance such as "給我介紹迪士尼樂園的詳細資料" *(Give me more details about Disneyland)*, which sets the concept value "detail=詳細 *(details)*", then the third rule condition in Table 8 is satisfied and Rule 3 will be used as the message plan.
- If the user requested an attraction which cannot be found in the database, then the second rule selection condition in Table 8 is satisfied and Rule 4 will be selected as the message plan. As a consequence, the system will apologize for not being able to offer relevant information.

4.2 Response Message Realization Using Corpus-Derived Templates

The semantic transitions above generates a message plan for generating the system response, in the form of semantic abstraction (SA) $\{TG, DA, KC\}_{\text{system}}$. Analysis of our regularized corpus also suggests that each of these SA may be verbalized in a variety of ways. These verbalization methods found in the corpus are encoded in a set of 89 message realization templates with labels, e.g. GENERAL_INFO, PICTURE_INFO, GOOD_TRIP, etc., as shown in Table 9.

Table 9. Examples of message realization templates derived from the regularized corpus

Text Generation Templates:

Template Label: GENERAL_INFO
Contents: 這裏是{request#attraction}的資料，請看。
(translation: here is the information about {request#attraction}).

Template Label: PICTURE_INFO
Contents: 我想你可以看看{request#attraction}的圖片資料{database#picture}
(translation: you may refer to these pictures {database#picture} of {request#attraction}).

Template Label: GOOD_TRIP
Contents: 祝你旅途愉快！
(translation: have a good trip)

A given $\{TG, DA, KC\}_{\text{system}}$ may correspond to one or more message realization templates. In cases where there are multiple options, we devise a set of template selection rule based on the regularized corpus. This is illustrated in Table 10, where the system response with SA *{ATTRACTİON, İNFORM_GENERAL, attraction}*$_{system}$ may be verbalized by the templates GENERAL_İNFO or PİCTURE_İNFO, depending on whether database retrieval can provide a picture, i.e. {database#picture}!=null). All system reponses with the dialog act of BYE, regardless of the task goal, will be realized by the template GOOD_TRİP.

Table 10. Illustration of a template selection rule among possible message realization templates that correspond to a given $\{TG, DA, KC\}_{\text{system}}$. The asterisk (*) is a wildcard that matches all task goals (TG).

Semantic Abstraction of the System's Response: {ATTRACTION, INFORM_GENERAL, attraction}system Associated Text Generation Templates: **Option 1**: GENERAL_INFO **Option 2**: PICTURE_INFO Template Selection Rule: IF ({grammar#picture}!=null) THEN select Option 2 ELSE select Option 1
Semantic Abstraction of the System's Response: {*, BYE, bye}system Associated Text Generation Templates: GOOD_TRIP

5 Evaluation

To evaluate the quality of responses generated by the NLG component, we recruited 15 subjects and asked them to play the role of a tourist in Hong Kong and make related inquiries. The subjects first attend a briefing session where they are presented with the knowledge scope of the system and the supported informational goals (i.e. the 12 task goals in Table 4). The subjects are then instructed to interact with the system textual input and output. The entire interaction is logged and the subjects are subsequently asked to refer to the logged responses (1230 in total) and evaluate each generated response in two ways:

(i) Task Completion Rate – A task is considered complete if the appropriate message exists in the response. For example, if the subject's question is: "迪士尼樂園的門票多少錢?" *(What is the price of a ticket for Disneyland?)* and the system's response is: "迪士尼樂園的票價為成人295，小童210，長者170" *(Ticket prices for Disneyland is 295 for adults, 210 for children and 170 for seniors)* – the response is considered complete. If the subject's question is: "迪士尼樂園的兒童票幾錢？" *(How much is children's ticket for Disneyland?)* and if the system provides the same answer, the task is also considered complete because the response contains the expected information "小童*210" (210 for children).* The specificity of the answer is dependent on the current design of the database. It is possible that more specific answers can be generated if the database supports finer granularities in knowledge engineering. Overall 83% of the generated response turns are considered relevant for the task goals based on the user's request turns.

(ii) Grice's Maxims and User Satisfaction – with reference to our previous work [13], we also conducted qualitative evaluation based on Grice's maxims [14] as well as overall user satisfaction. The qualitative evaluation uses a five-point Likert scale (very poor / poor / average / good / very good). Each subject is asked rate the overall quality of the generated responses during his/her interaction with the system, by answering the following questions in a questionnaire:

- **Maxim of Quality**, i.e. system responses should be true with adequate evidence - "*Do you think that the answers are accurate and true?*"
- **Maxim of Quantity**, i.e. system should give sufficient information - "*Do you think that the answers are informative?*"
- **Maxim of Relevance**, i.e. system responses should be relevant to the ongoing conversation - "*Do you think that the answers are relevant to the conversation?*"
- **Maxim of Manner**, i.e. system responses should be brief and clear, with no obscurity or ambiguity - "*Do you think that the answers are clear?*"
- **Overall User Satisfaction** - "*To what extent are you satisfied with the overall performance of the system in responding to your questions?*"

Table 11 shows the average scores and standard derivations (in brackets) of the evaluation results. A *t*-test shows that our results are significantly better than average (Likert score 3) at α=0.06.

Table 11. Evaluation results of our response generation system in terms of Grice's Maxims and user satisfaction

Quality	Quantity	Relevance	Manner	Satisfaction
4.0 (0.7)	4.1 (0.8)	3.8 (0.7)	3.9 (0.8)	4.0 (0.6)

Analysis of the evaluation logs indicated one common error which accounted to 10% of the incomplete tasks. For example, we found that if the discourse history involved an inquiry with the task goal (TG) of ROUTE, as in the question "怎麼去海洋公園？" *(How to I get to Ocean Park?)* followed by a general question that does not have an obvious TG, e.g. "海洋公園有什麼？" *(What's there in Ocean Park?)* ; then the discourse inheritance mechanism will inherit the TG of ROUTE to the current question, thereby leading to the generation of an incoherent response. Based on the comments offered by the subjects after the evaluation exercise, this kind of error was the main cause of dissatisfaction during the interaction.

6 Conclusions and Future Work

This paper presents a corpus-based approach for cooperative response generation in a spoken dialog system for the Hong Kong tourism domain. A corpus with 3874 requests and responses is collected using Wizard-of-Oz framework. The corpus then undergoes a regularization process that simplifies the interactions to ease subsequent modeling. A semi-automatic process is developed to annotate the each utterance in the dialog turns in terms of their key concepts (KC), task goal (TG) and dialog acts (DA). TG and DA characterize the informational goal and communicative goal of the utterance respectively. The annotation procedure is integrated with a dialog modeling heuristic and a discourse inheritance strategy to generate a semantic abstraction (SA), in the form of $\{TG, DA, KC\}$, for each user request and system response in the dialog. Semantic transitions, i.e. $\{TG, DA, KC\}_{\text{user}} \rightarrow \{TG, DA, KC\}_{\text{system,}}$ may hence be directly derived from the corpus as rules for *response message planning*. Related verbalization methods may also be derived from the corpus and used as templates for *response message realization*. All the rules and templates are stored externally in a human-readable text file which brings the advantage of easy extensibility of the system. Evaluation of this corpus based approach shows that 83% of the generated responses are coherent with the user's request and qualitative rating achieves a score of 4.0 on a five-point Likert scale. Future work will be devoted towards response generation of semantic-dependent expressive markups for text-to-speech synthesis.

Acknowledgments. This work is support by the research fund from the Research Grant Council of Hong Kong (RGC) under grant No. CUHK4237/03E and the joint fund of NSFC (National Natural Science Foundation of China) - RGC under grant No. 60418012 and N-CUHK417/04.

References

1. Hovy, E.: Language Generation. Survey of the State of the Art in Human Language Technology (1996)
2. Bateman, J., Henschel, R.: From Full Generation to "Near-Templates" without losing generality. In: Proc. of the KI'99 Workshop, "May I Speak Freely?" (1999)
3. Heisterkamp, P.: Time to Get Real: Current and Future Requirements for Generation in Speech and Natural Language from an Industrial Perspective. In: Proc. of the KI'99 Workshop, "May I Speak Freely?" (1999)
4. Bateman, J.: KPML Development Environment: Multilingual Linguistic Resource Development and Sentence Generation. German National Center for Information Technology, IPSI (1997)
5. Elhadad, M., Robin, J.: An Overview of SURGE: A Reusable Comprehensive Syntactic Realization Component. Technical Report 96-03, Dept. of Mathematics and Computer Science, Ben Gurion University (1996)
6. Galley, M., Fosler-Lussier, E., Potamianos, A.: Hybrid Natural Language Generation for Spoken Dialogue Systems. In: Proc. of Seventh European Conference on Speech Communication and Technology (Eurospeech '01), Aalborg, Denmark (2001)
7. Young, S.: Talking to Machines (Statistically Speaking). In: Proc. of the International Conference on Spoken Language Processing (2002)
8. Walker, M., Rambow O., Rogati, M.: A Trainable Approach to Sentence Planning for Spoken Dialogue. Computer Speech and Language (2002)
9. Alexandersson, J., Buschbeck-Wolf, Fujinami, M.K., Koch, E.M., Reithinger, B.S.: Acts in VERBMOBIL-2 Second Edition: Verbmobil Report 226, Universitat Hamburg, DFKI Saarbrucken, Universitat Erlangen, TU Berlin

10. Allen, J., Core, M.: Draft of DAMSL: Dialog Act Markup in Several Layers, http://www.cs.rochester.edu/research/speech/damsl/RevisedManual/RevisedManual.html
11. Meng, H., Lam, W., Wai, C.: To Believe is to Understand. In: Proc. of Eurospeech (1999)
12. Chan, S.F., Meng, H.: Interdependencies among Dialog Acts, Task Goals and Discourse Inheritance in Mixed-Initiative Dialogs. In: Proc. of Human Language Technology (2002)
13. Meng, H., Yip, W.L, Mok, O.Y., Chan, S.F.: Natural Language Response Generation in Mixed-Initiative Dialogs using Task Goals and Dialog Acts. In: Proc. of Eurospeech (2003)
14. Frederking, R.: Grices's Maxims: Do the Right Thing. Frederking, R.E. (1996)

A Cantonese Speech-Driven Talking Face Using Translingual Audio-to-Visual Conversion

Lei Xie[1], Helen Meng[1], and Zhi-Qiang Liu[2]

[1] Human-Computer Communications Laboratory
Dept. of Systems Engineering & Engineering Management
The Chinese University of Hong Kong, Hong Kong
{lxie, hmmeng@}@se.cuhk.edu.hk
[2] School of Creative Media
City University of Hong Kong, Hong Kong
zq.liu@cityu.edu.hk

Abstract. This paper proposes a novel approach towards a video-realistic, speech-driven talking face for Cantonese. We present a technique that realizes a talking face for a target language (Cantonese) using only audio-visual facial recordings for a base language (English). Given a Cantonese speech input, we first use a Cantonese speech recognizer to generate a Cantonese syllable transcription. Then we map it to an English phoneme transcription via a translingual mapping scheme that involves symbol mapping and time alignment from Cantonese syllables to English phonemes. With the phoneme transcription, the input speech, and the audio-visual models for English, an EM-based conversion algorithm is adopted to generate mouth animation parameters associated with the input Cantonese audio. We have carried out audio-visual syllable recognition experiments to objectively evaluate the proposed talking face. Results show that the visual speech synthesized by the Cantonese talking face can effectively increase the accuracy of Cantonese syllable recognition under noisy acoustic conditions.

1 Introduction

With the recent advances in multimedia technologies, animated characters, such as talking faces/heads, are playing an increasingly important role in human-computer communication. Talking faces can be driven by input text or input speech[1]. While text-driven talking faces employ both synthesized voices and faces, constituting text-to-audiovisual speech (TTAVS); speech-driven talking faces involve synthesizing visual speech information from real speech. A speech-driven talking face may serve as an aid to the hearing-impaired as the visual speech signal can effectively augment the audio speech signal (eg. by lip-reading) in order to enhance clarity in speech perception. The timing information needed for visual speech synthesis must be synchronized to the input audio speech signal. Such timing information may be obtained by means of a speech recognizer. Hence, speech-driven talking face synthesis is an interesting and feasible research problem [1].

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 627–639, 2006.

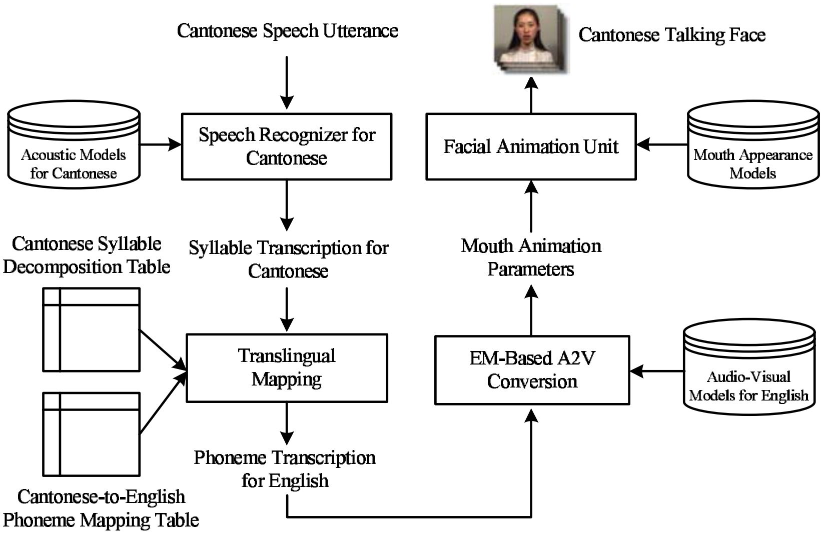

Cantonese Speech Utterance
Acoustic Models for Cantonese
Speech Recognizer for Cantonese
Syllable Transcription for Cantonese
Cantonese Syllable Decomposition Table
Translingual Mapping
Cantonese-to-English Phoneme Mapping Table
Phoneme Transcription for English
EM-Based A2V Conversion
Audio-Visual Models for English
Mouth Animation Parameters
Facial Animation Unit
Mouth Appearance Models
Cantonese Talking Face

Fig. 1. Block Diagram of the Cantonese Talking Face System

During the last decade, various talking faces have been proposed, pursuing either a natural 3D facial mesh [2] or video-realistic effects [3]. These talking faces are mostly driven by English phonetics (or visemes). Recently we also see talking faces driven by Finnish [4], Italian [5], Chinese Mandarin (Putonghua) and Cantonese [6]. A related problem is how to animate a talking face designed based on phonetics in one language, with input audio speech in another (target) language. For example, Verma *et al.* [7] have proposed a Hindi talking face based on a translingual mapping between Hindi and English phonemes. In this paper, we extend our previous work on an English talking face [8], such that it may be driven by input Cantonese speech. Such translingual audio-visual associations enhance the inter-operability between audio speech analysis and visual speech synthesis.

The rest of the paper is organized as follows. The following section describes the block diagram of our talking face system. In Section 3, the translingual audio-to-visual conversion scheme is presented in detail. Section 4 describes our facial animation unit. In Section 5, experiments are carried out to evaluate our Cantonese talking face. Finally conclusions are drawn in Section 6.

2 System Overview

Fig. 1 shows the block diagram of the proposed Cantonese talking face system. The system is composed of four main phases—a Cantonese speech recognizer, a translingual mapping unit, an expectation maximization (EM)-based audio-to-visual (A2V) converter and a facial animation unit.

The initial audio-visual model is developed based on English phonetics. Input English audio speech is fed into the A2V converter which generates mouth animation parameters. This A2V converter adopts an EM-based conversion algorithm which generates mouth parameters frame by frame under the maximum likelihood (ML) sense, which is frame-synchronized to the audio input. These generated mouth images are "stitched" onto a background facial image sequence using a facial animation unit. Since this work presents a Cantonese speech-driven talking face, we need to extend the existing framework to cover the target language of Cantonese, as described below.

Different from our previous English talking face, in this work we use a Cantonese speech recognizer to generate a Cantonese syllable transcription for the input audio. Subsequently, the translingual mapping unit is in charge of mapping the Cantonese syllable transcription into a reasonable English phoneme transcription where each phonetic unit is associated with estimated timing information. As the initializations, the corresponding visual model means associated with the English phonetic string, together with the input Cantonese audio, is fed into the A2V converter.

3 Translingual Audio-to-Visual Conversion

We have developed a translingual audio-to-visual conversion scheme that is capable of converting speech input in the target language (namely Cantonese) into mouth animation parameters corresponding to the base language (i.e., English) of the existing audio-visual model. This facilitates inter-operability between the audio speech analysis component and the visual speech synthesis component. In this way, we do not need to record a new visual database for visual speech synthesis.

3.1 Audio-Visual Modelling in the Base Language of English

English is the base language of our audio-visual model since we have already proposed a video-realistic talking face [8] that learned audio-visual associations for spoken English from audio-visual facial recordings. These facial recordings involve head-and-shoulder front-view videos of a female speaker uttering 524 TIMIT sentences.[1] Each acoustic feature vector includes 12 MFCCs with log energy and their first and second order derivatives (hence 39 dimensions in total). The mouth region-of-interest (ROI) was first tracked, and encoded using the principal component analysis (PCA). To achieve video-realistic animation, we used PCA to get the visual features that capture mouth appearance in a low dimension (30 PCA coefficients here).

We used multi-stream hidden Markov models (MSHMMs) [9] to model the audio-visual articulation process in terms of context-dependent (CD) phoneme

[1] For details on the AV recordings, please refer to http://www.cityu.edu.hk/rcmt/mouth-synching/jewel.htm

Table 1. Cantonese phonetic decomposition table (partial)

		Initial	Nucleus	Coda	E.g. Character
uk	phn. string	×	*u*	*k*	屋
	duration	0	0.5	0.5	
bing	phn. string	*b*	*i*	*ng*	並
	duration	0.2	0.4	0.4	
baang	phn. string	*b*	*aa*	*ng*	崩
	duration	0	0.5	0.5	
loeng	phn. string	*l*	*eo*	*ng*	涼
	duration	0.3	0.35	0.35	
jyun	phn. string	*j*	*yu*	*n*	員
	duration	0.25	0.375	0.375	

Note: '×' denotes a NULL phoneme.

models (triphones and biphones). We used two-stream, state-synchronous MSHMMs in audio-visual modelling, where two observation streams are incorporated to describe audio and visual modalities respectively. In its general form, the class conditional observation likelihood of the MSHMM is the product of the observation likelihoods of its single-stream components, where stream exponents are used to capture the reliability of each modality.

Given the bimodal observation $\mathbf{o}_t^{av} = [\mathbf{o}_t^a, \mathbf{o}_t^v]$ at frame t, the state emission likelihood of a MSHMM is

$$P(\mathbf{o}_t^{av}|c) = \prod_{s\in\{a,v\}} \left[\sum_{k=1}^{K_{sc}} \omega_{sck}\mathcal{N}_s(\mathbf{o}_t^s;\mu_{sck},u_{sck})\right]^{\lambda_{sct}}, \quad \sum_s \lambda_{sct} = 1 \tag{1}$$

where λ_{sct} denotes the stream exponents, which are non-negative, and a function of modality s, the HMM state c, and frame t. The state dependence is to model the local, temporal reliability of each stream. We set $\lambda_{sct} = 0.5$ for all s, c and t supposing audio speech and visual speech have the same contribution. $\mathcal{N}_s(\mathbf{o}_t^s;\mu_{sck},u_{sck})$ is the Gaussian component for state c, stream s, and mixture component k with mean μ_{sck} and covariance u_{sck}. In total we trained 423 MSHMMs for triphones, biphones and monophones. Each MSHMM has 3 emitting states with 6 continuous density Gaussian mixtures.

3.2 Translingual Mapping

The current work aims to integrate Cantonese audio speech analysis and English visual speech synthesis. This involves a translingual mapping of two levels:

- **Symbols:** Different languages have different phonological units, e.g. syllables are commonly used for Cantonese and phonemes are commonly used for English. This also entails different contextual representations, e.g. initial-finals for Cantonese and triphones/biphones for English. The different symbolic representations need to be bridged.

- **Timing:** Phonetic units of different languages may have different time durations. The audio frame rates used in the recognizer may be different from the video frame rate of the audio-visual models. Therefore, time alignments must be considered along with the mapping across symbolic representations.

3.2.1 Mapping Across Different Symbolic Representation Systems

Previous work in Translingual speech-driven talking face has involved Hindi and English[7]. These two languages are both Indo-European, and can be accomplished by simple phoneme-to-phoneme mapping. Our approach involves mapping between Chinese and English that their phonological architectures are quite different [10].

The Chinese spoken languages (e.g. Cantonese) do not have explicit word delimiters and a word may contain one or more characters. Each character is pronounced as a *syllable*, and an utterance is heard as a string of momosyllabic sounds with tones. If we ignore the tonal variations, the syllable unit is commonly referred to as a *base syllable*. In general, there are about 600 base syllables in Cantonese. Each base syllable is decomposed into an *initial* and a *final*, and a final can be further subdivided into a *nucleus* and a *coda*. For example, the syllable /nei/( ) is composed of a initial /n/, a nucleus /ei/ and a null coda. In Cantonese, there are about 20 initials and 53 finals. If we categorize these units (initials, nucleus, and codas) with the same (or similar) pronunciations into a phonetic class, there are altogether about 28 "phonetic" classes. Recall that this work needs to map symbolic representation of Cantonese phonetics to that of English phonetics. Based on the above phonetic classifications, our approach involves the following two steps.

- Decompose a base syllable into a sub-syllable string with an initial, a nucleus and a coda, which constitutes a Cantonese "phonetic" string;
- Map the Cantonese "phonetic" string to an English phonetic string via a translingual mapping table.

Table 1 and Table 2 show fragments of the Cantonese phonetic decomposition table and the translingual mapping table respectively. Note that the phoneme durations in Table 1 are obtained from Cantonese syllable samples, and also some Cantonese phonemes are mapped to English phoneme pairs in Table 2. For example, /yu/ is mapped to {/ih/, /uw/}.

In our approach, we used a homegrown Cantonese base syllable recognizer [11] to transcribe input Cantonese speech. In this recognizer, the acoustic models includes three-state HMMs for syllable initials and five-state HMMs for syllable finals. These acoustic models are context-dependent HMMs, namely *initial-final* models, with 16 Gaussian mixtures. They were initially trained with clean, read speech from CUSENT, [2] and then adapted with studio anchor speech recorded

[2] http://dsp.ee.cuhk.edu.hk/speech/cucorpora/

Table 2. Cantonese-to-English phoneme mapping table (partial)

Cantonese Phoneme	E.g.	English Phoneme	E.g.
b	**b**ou(報)	*b*	**b**oy
m	**m**ei(美)	*m*	li**m**it
h	**h**aa(下)	*hh*	**h**air
gw	**gw**ok(國)	*g-w*	**g**row-al**w**ays
i	s**i**(士)	*ih*	h**i**lls
oe	j**oe**ng (洋)	*er*	m**ur**der
yu	j**yu**n (員)	*ih-uw*	h**i**lls-t**wo**

from the news broadcasts of the Hong Kong TVB Jade Channel. (about 40 minutes). The syllable recognition accuracy is 59.3%. Further details on the recognizer can be found in [11].

We first collected the base syllable transcription for a Cantonese utterance, and subsequently aligned the transcription to initial-final symbols via the Viterbi algorithm [9]. The core sub-syllables, e.g., /F_ei/ in /I_g−F_ei+I_gw/,[3] were mapped to English phonemes via Table 1 and Table 2, as illustrated in Fig. 2 (a) and (b). Since we used context-dependent AV models (triphone and biphone MSHMMs) to catch the coarticulation phenomena, we further expanded the English phonemes to triphones or biphones by considering the nearest neighbors, as shown in Fig. 2 (d). The triphones and biphones were selected from the 423 AV models. If a triphone (or biphone) match cannot be found in the model list, a simple phoneme model is chosen.

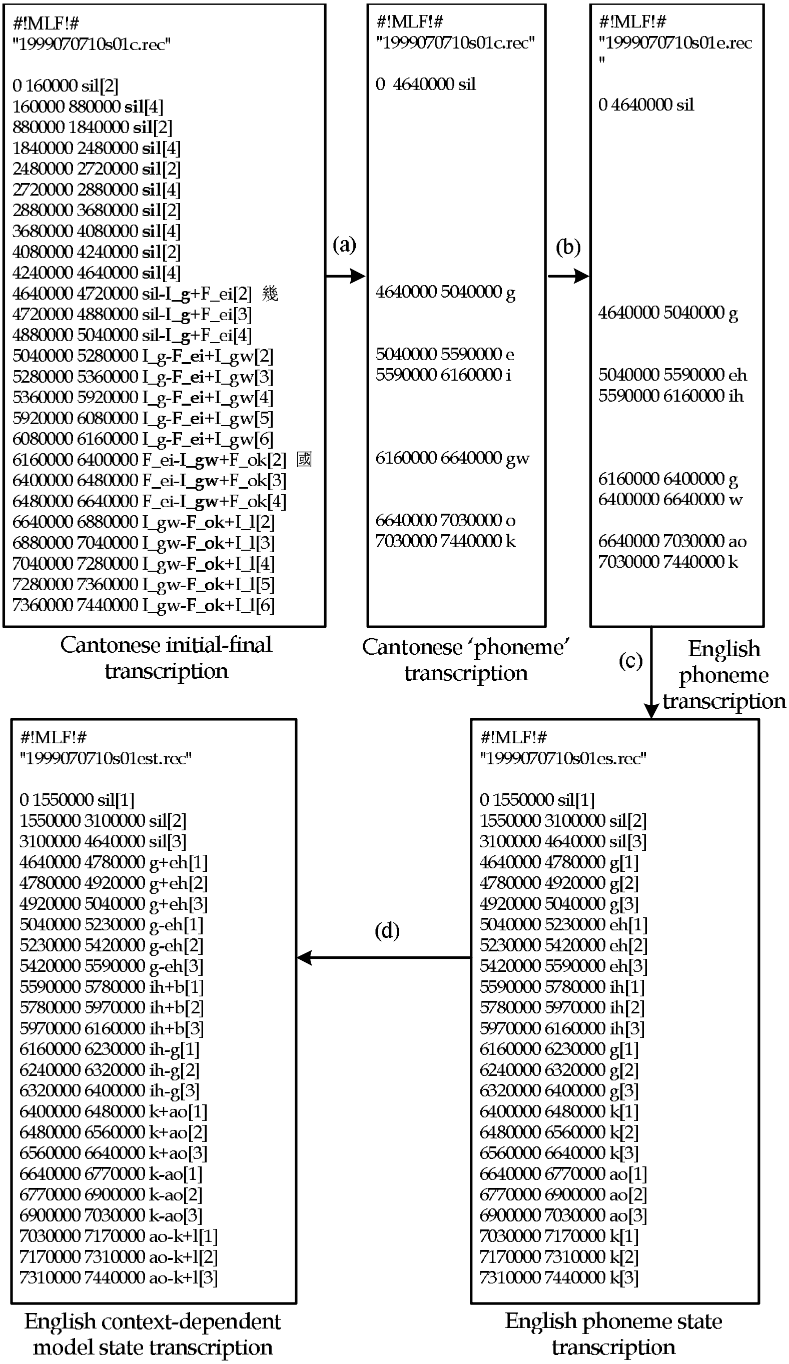
3.2.2 Time Alignment

Previous research have shown that humans are quite sensitive to the timing relations between audio and visual speech [3]. Therefore, we use the following steps to capture reasonable time relations:

Step 1: If the sub-syllable is an initial (I_*), its duration is directly obtained from the alignment of the speech recognition against the recognized sub-syllable units. If the sub-syllable unit is a final (F_*), the durations of its nucleus and coda are assigned via the durations defined in Table 1. Note that state durations are merged to the model level. For example in Fig. 2 (a), the durations of initials /g/ and /gw/ are directly obtained from the alignment result. The durations of Cantonese "phonemes" /e/ and /i/ are obtained from Table 1. The durations of the three states of /F_ei−I_gw+F_ok/ are merged.

Step 2: Cantonese "phoneme" durations are directly assigned to English phoneme durations. If the Cantonese "phoneme" is mapped to an English phoneme pair, the duration of each English phoneme is a half duration of the Cantonese "phoneme". For example in Fig. 2 (b), the durations of /g/, /eh/, /ih/, /ao/ and /k/ are directly obtained from /g/, /e/, /i/, /o/ and /k/, while the durations of /g/ and /w/ are half durations of /gw/.

[3] Where 'I' denotes Cantonese syllable initial, and 'F' denotes syllable final.


Cantonese initial-final transcription
(a)
Cantonese ‘phoneme’ transcription
(b)
English phoneme transcription
(c)
English phoneme state transcription
(d)
English context-dependent model state transcription

```
#!MLF!#
"1999070710s01c.rec"

0 160000 sil[2]
160000 880000 sil[4]
880000 1840000 sil[2]
1840000 2480000 sil[4]
2480000 2720000 sil[2]
2720000 2880000 sil[4]
2880000 3680000 sil[2]
3680000 4080000 sil[4]
4080000 4240000 sil[2]
4240000 4640000 sil[4]
4640000 4720000 sil-I_g+F_ei[2] 幾
4720000 4880000 sil-I_g+F_ei[3]
4880000 5040000 sil-I_g+F_ei[4]
5040000 5280000 I_g-F_ei+I_gw[2]
5280000 5360000 I_g-F_ei+I_gw[3]
5360000 5920000 I_g-F_ei+I_gw[4]
5920000 6080000 I_g-F_ei+I_gw[5]
6080000 6160000 I_g-F_ei+I_gw[6]
6160000 6400000 F_ei-I_gw+F_ok[2] 國
6400000 6480000 F_ei-I_gw+F_ok[3]
6480000 6640000 F_ei-I_gw+F_ok[4]
6640000 6880000 I_gw-F_ok+I_l[2]
6880000 7040000 I_gw-F_ok+I_l[3]
7040000 7280000 I_gw-F_ok+I_l[4]
7280000 7360000 I_gw-F_ok+I_l[5]
7360000 7440000 I_gw-F_ok+I_l[6]
```

```
#!MLF!#
"1999070710s01c.rec"

0  4640000 sil

4640000 5040000 g

5040000 5590000 e
5590000 6160000 i

6160000 6640000 gw

6640000 7030000 o
7030000 7440000 k
```

```
#!MLF!#
"1999070710s01e.rec
"

0 4640000 sil

4640000 5040000 g

5040000 5590000 eh
5590000 6160000 ih

6160000 6400000 g
6400000 6640000 w

6640000 7030000 ao
7030000 7440000 k
```

```
#!MLF!#
"1999070710s01es.rec"

0 1550000 sil[1]
1550000 3100000 sil[2]
3100000 4640000 sil[3]
4640000 4780000 g[1]
4780000 4920000 g[2]
4920000 5040000 g[3]
5040000 5230000 eh[1]
5230000 5420000 eh[2]
5420000 5590000 eh[3]
5590000 5780000 ih[1]
5780000 5970000 ih[2]
5970000 6160000 ih[3]
6160000 6230000 g[1]
6240000 6320000 g[2]
6320000 6400000 g[3]
6400000 6480000 k[1]
6480000 6560000 k[2]
6560000 6640000 k[3]
6640000 6770000 ao[1]
6770000 6900000 ao[2]
6900000 7030000 ao[3]
7030000 7170000 k[1]
7170000 7310000 k[2]
7310000 7440000 k[3]
```

```
#!MLF!#
"1999070710s01est.rec"

0 1550000 sil[1]
1550000 3100000 sil[2]
3100000 4640000 sil[3]
4640000 4780000 g+eh[1]
4780000 4920000 g+eh[2]
4920000 5040000 g+eh[3]
5040000 5230000 g-eh[1]
5230000 5420000 g-eh[2]
5420000 5590000 g-eh[3]
5590000 5780000 ih+b[1]
5780000 5970000 ih+b[2]
5970000 6160000 ih+b[3]
6160000 6230000 ih-g[1]
6240000 6320000 ih-g[2]
6320000 6400000 ih-g[3]
6400000 6480000 k+ao[1]
6480000 6560000 k+ao[2]
6560000 6640000 k+ao[3]
6640000 6770000 k-ao[1]
6770000 6900000 k-ao[2]
6900000 7030000 k-ao[3]
7030000 7170000 ao-k+l[1]
7170000 7310000 ao-k+l[2]
7310000 7440000 ao-k+l[3]
```

Fig. 2. An example of the translingual mapping process. Label format: start_time end_time phonetic_label[state].

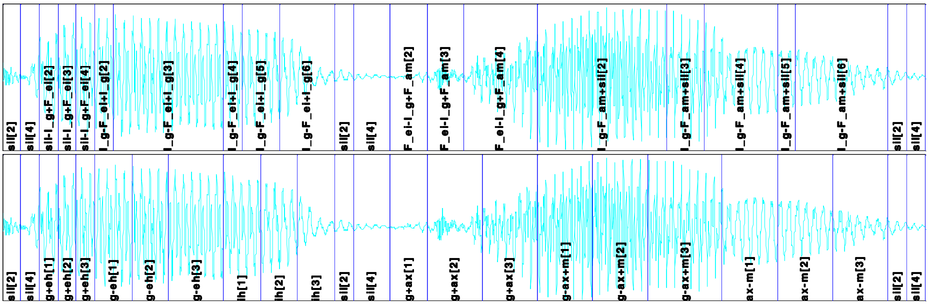


Fig. 3. Time alignment result for a speech fragment. Up: Cantonese initial-final transcription, Bottom: English context-dependent model state transcription.

Step 3: The duration of a state is 1/3 of that of a phoneme (or triphone, biphone). For example in Fig. 2 (c), the duration of state /g[1]/ is 1/3 of that of /g/.

Note that we directly use the the durations for initials generated from the recognizer in Step 1 since they are more accurate for the specific utterance as compared to the statistics from syllable samples. Fig. 3 shows a time alignment result for a Cantonese speech fragment using the above steps.

Finally, the average values of visual Gaussian means associated with each model states

$$\mathbf{o}_t^v = \sum_k w_{\theta_t^v k} \mu_{\theta_t^v k} \tag{2}$$

were used as the initializations of mouth animation parameters, where θ_t^v is the mapped phonetic model state at t. These initialized values were fed into the EM-based A2V converter.

3.3 EM-Based AV Conversion

We used an EM-based audio-to-visual conversion method [8] which directly resulted in mouth parameters (i.e., estimated PCA coefficients) framewise under the ML criterion. The EM-based conversion method has been shown robust to speech degradations, resulting in decent mouth parameters [8].

Given the input audio data $\mathbf{O}^a$ and the trained MSHMMs λ, we seek the missing visual observations (i.e. parameters) $\hat{\mathbf{O}}^v$ by maximizing the likelihood of the visual observations. According to the EM solution of ML, we maximize an auxiliary function:

$$\hat{\mathbf{O}}^v = \arg \max_{\mathbf{O}^{v'} \in \mathcal{O}^v} \mathcal{Q}(\lambda, \lambda; \mathbf{O}^a, \mathbf{O}^v, \mathbf{O}^{v'}), \tag{3}$$

where $\mathbf{O}^v$ and $\mathbf{O}^{v'}$ denote the old and new visual observation sequences in the visual observation space $\mathcal{O}^v$ respectively.

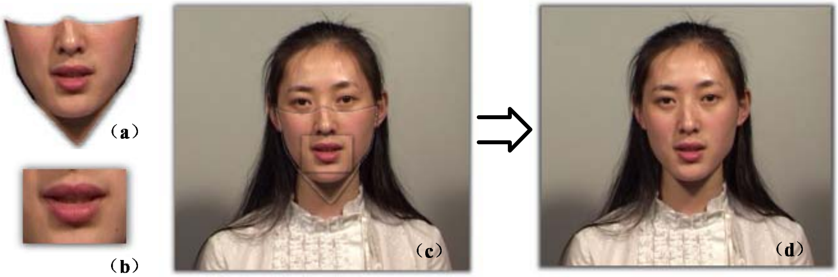


Fig. 4. The three-layer overlay process. (a) a jaw candidate, (b) a synthesized mouth, (c) stitching to face and (d) a resultant frame.

By taking derivative of $\mathcal{Q}(\lambda, \lambda; \mathbf{O}^a, \mathbf{O}^v, \mathbf{O}^{v'})$ respect to $\mathbf{o}_t^{v'}$ to zero, we get [8]

$$\hat{\mathbf{o}}_t^v = \frac{\sum_{q_t} \sum_k \gamma_t(q_t, k) \omega_{q_t vk} u_{q_t vk}^{-1} \mu_{q_t vk}}{\sum_{q_t} \sum_k \gamma_t(q_t, k) \omega_{q_t vk} u_{q_t vk}^{-1}}, \tag{4}$$

where q_t is the possible state of t, and the *occupation* probabilities $\gamma_t(q_t, k)$ can be computed using the forward-backward algorithm described in the E-Step of the EM algorithm. Since the EM algorithm converges to a local minimum, a good parameter initialization is essential for accurate mouth parameters. Therefore, we adopted the visual Gaussian means associated with the mapped English phonetic transcriptions (see Eq. (2)) as the initializations.

4 Video-Realistic Facial Animation

The facial animation unit first smoothes the estimated mouth parameters (i.e., PCA coefficients) by a moving average filter (width=3) to remove possible jitters, and then augments the fine appearance details through a performance refinement process indicated in [12]. Mouth images are generated from the estimated PCA coefficients by the PCA expansion process. Finally, the synthesized mouth frames are overlaid onto a base facial video clip.

We used a three-layer overlaying process (see Fig. 4), where the synthesized mouth, the corresponding jaw, and the face background snippet are sewed up by the Poisson image editing technique [14]. We associated an appropriate jaw from a jaw candidate set to each synthesized mouth according to the mouth opening scale and the waveform energy [12]. To avoid jerky animation induced by stitching coordinates errors, we used a facial feature tracking method [3] with sub-pixel accuracy. Fig. 5 illustrates some snapshots from a synthesized talking face video clip.

5 Evaluations

To evaluate the proposed Cantonese talking face, we carried out objective evaluations using audio-visual speech recognition (AVSR) experiments. This kind

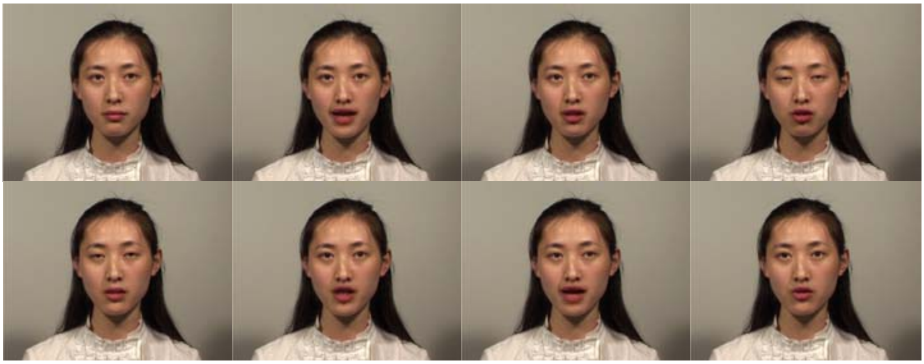


Fig. 5. Some snapshots from a synthesized video

Table 3. Evaluation systems

System	Features & Models	Training & Testing
AO	MFCCs+Δ+Δ^2 (39); CD-HMMs with 16 mixtures	*Training*: Original audio (40 mins); *Testing*: Original, 20db, 10dB (207 secs)
AV-*nontrans*	Audio: MFCCs+Δ+Δ^2 (39); Video: PCA Coefs. (30); CD-MSHMMs with 16 mixtures for audio and 6 mixtures for video; Without translingual mapping	*Training (Audio)*: Original audio (40 mins); *Testing (Audio)*: Original, 20db, 10dB (207 secs)
AV-*trans*	Audio: MFCCs+Δ+Δ^2 (39); Video: PCA Coefs. (30); CD-MSHMMs with 16 mixtures for audio and 6 mixtures for video; With translingual mapping	*Training (Video)*: Estimated PCA Coefs. from original audio (40 mins); *Testing (Video)*: Estimated PCA Coefs. from originial audio (207 secs)

of lipreading test by machine was used to evaluate the quality of the mouth animation (i.e. visual speech) in terms of the improvement in speech recognition accuracy of an AVSR system versus an audio-only ASR system. It provides a way to evaluate the quality of visual speech synthesis by means of machine perception.

5.1 Experiment Setup

We used the hand-transcribed anchor speech (about 40 minutes) from the Cantonese news broadcasts of the Hong Kong TVB Jade channel (described in Section 3.2) as the training data, and another 207 seconds anchor speech were used as the testing set. Speech babble noise (simultaneous speech from multiple speakers collected from cafeteria environment) was added to the testing speech at two signal-noise-ratio (SNR) conditions (20dB and 10dB). As a sanity check, we also developed a talking face without the translingual mapping, where Cantonese

Table 4. Experimental results

	AO	AV-*nontrans*	AV-*trans*
Ori.	59.3	59.4	59.6
20*d*B	40.6	46.2	50.0
10*d*B	19.0	28.8	34.3

input speech was directly converted to an English phonetic transcription by an English recognizer. The English recognizer was trained using the audio data from the English audio-visual facial recordings described in Section 3.1. We carried out syllable recognition experiments, and collected syllable accuracy rates for an audio-only ASR system and two AVSR systems. In the AVSR systems, we also adopted the state-synchronous context-dependent MSHMMs described in Section 3.1 as the audio-visual modelling scheme, and the estimated animation parameters (i.e., PCA coefficients) from the original speech were used as the visual features. The stream exponents were selected by minimizing the syllable error rate.

In the experiments, we also used the Cantonese syllable recognizer described in Section 3.2 as the audio-only (AO) baseline system to benchmark the test. The AO system was trained using the same training data. Experiments were performed under mismatched training-testing conditions, i.e., the recognizer was trained using original clean speech, while tested using contaminated speech (10*d*B and 20*d*B SNR). Table 3 summarizes the system configurations.

5.2 Experimental Results

From results in Table 4, we can clearly observe that the AO system is heavily affected by additive noise. When the SNR is decreased to 10*d*B, the syllable accuracy is only 19.0%. The insertion errors contribute a lot to the accuracy decrease. This also shows that training-testing mismatch can drastically affect the performance of a recognizer. Not surprisingly, with the help of the visual speech information provided by the talking faces, both the AV-*nontrans* and the AV-*trans* systems significantly improve the accuracy rates at noisy conditions, with the latter (with the translingual mapping) being superior, yielding a 3.8% and a 5.5% absolute accuracy increase at 20*d*B and 10*d*B SNR respectively as compared with the former (without the translingual mapping). These promising results show that the visual speech synthesized by the proposed talking face contains useful lipreading information that can effectively increase the accuracy of machine speech perception under noisy conditions.

6 Conclusions

This paper presents a video-realistic, speech-driven talking face for Cantonese using only audio-visual facial recordings for English. We have developed a translingual audio-to-visual conversion scheme, which is composed of a Cantonese speech recognizer, a translingual mapping scheme and an EM-based audio-to-visual converter. The translingual mapping involves symbol mapping and also time alignment from Cantonese syllables to English phonemes. With the help of the translingual audio-to-visual conversion scheme, Cantonese speech is converted to mouth animation parameters using audio-visual English phonetic models. The mouth parameters are resembled to mouth images, and stitched onto a background facial image sequence. We have demonstrated that the visual speech synthesized by the proposed Cantonese talking face can effectively improve the syllable recognition accuracy of machine speech perception under noisy acoustic conditions, for example improving the syllable accuracy rate from 19.0% to 34.3% at $10dB$ SNR.

The promising results in this work have shown that given recorded facial video clips for one language, it is possible to synthesize reasonable facial animation with speech from another language. Since perceptual evaluations by human viewers are more appropriate for visual speech synthesis, we are currently performing subjective evaluations.

Acknowledgements

We would like to thank the Television Broadcasts Limited for providing the Cantonese news audio. We gratefully acknowledge Ms. Ye Zhu who has done a lot in recording the facial videos. We also would like to thank Ms. Pui Yu Hui for providing the Cantonese syllable recognizer and her help on the translingual mapping. This work is partially funded by the CUHK Shun Hing Institute of Advanced Engineering affiliated in the Microsoft-CUHK Joint Lab. for Human-Centric Computing & Interface Technologies.

References

1. Ostermann, J., Weissenfeld, A.: Talking Faces–Technologies and Applications. Proc. 17th ICPR (2004)
2. Pighin, F., Hecker, D., Lischinski, R., Szeliski, D. H.: Synthesizing Realistic Facial Expressions from Photographs. Siggraph (1998) 75–84
3. Cosatto, E., Ostermann, J.: Lifelike Talking Faces for Interactive Services. Proceedings of IEEE. **91**(9) (2003) 1406–1429
4. Olives, J.-L., Sams, M., Kulju, J., Seppaia, O., Karjalainen, M., Altosaar, T., Lemmetty, S., Toyra, K., Vainio M.: Towards a High Quality Finnish Talking Head. IEEE 3rd Workshop on Multimedia Signal Processing (1999) 433–437
5. Pelachaud, C. E., Magno-Caldognetto, Zmarich, C., Cosi, P.: Modelling an Italian Talking Head. Proc. Audio-Visual Speech Processing (2001) 72–77
6. Wang, J.-Q., Wong, K.-H., Heng, P.-A., Meng, H., Wong, T.-T.: A Real-Time Cantonese Text-To-Audiovisual Speech Synthesizer. Proc. ICASSP (2004) 653–656
7. Verma, A., Subramaniam, V., Rajput, N., Neti, C.: Animating Expressive Faces Across Languages. IEEE Trans. on Multimedia. **6**(6) (2003) 791–800
8. Xie, L., Liu, Z.-Q.: An Articulatory Appraoch to Video-Realistic Mouth Animation. Proc. of ICASSP (2006) 593–596

9. Young, S., Evermann, G., Kershaw, D., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: The HTK Book (for HTK Version 3.2). Cambridge University Engineering Department (2002), [online] http://htk.eng.cam.ac.uk/
10. Linguistic Society of Hong Kong. Cantonese Transcription Scheme (1997)
11. Hui, P. Y., Lo, W. K., Meng, H.: Tow Robust Methods for Cantonese Spoken Document Retrieval. Proc. of 2003 ISCA Workshop on Multilingual Spoken Document Retrieval (2003) 7–12
12. Xie, L., Liu, Z.-Q.: A Coupled HMM Approach to Video-Realisic Speech Animation. Pattern Recognition, submitted (2006)
13. Cosatto, E.: Sample-Based Talking-Head Synthesis. Ph.D Thesis of Swiss Federal Institue of Technology (2002)
14. Pérez, P., Gangnet, M., Blake, A.: Poisson Image Editing. Siggraph (2003) 313–318

The Implementation of Service Enabling with Spoken Language of a Multi-modal System Ozone*

Sen Zhang[1] and Yves Laprie[2]

[1] The Graduate School, Chinese Academy of Sciences, Beijing
[2] Speech Group, INRIA-LORIA B.P.101 54602 Villers les Nancy, France
zhangsen@yahoo.com

Abstract. In this paper we described the architecture and key issues of the service enabling layer of a multi-modal system Ozone which is oriented for new technologies and services for emerging nomadic societies. The main objective of the Ozone system is to offer a generic framework to enable consumer-oriented Ambient-Intelligence applications. As a large multi-modal system, Ozone consists of many functional modules. However, spoken language played an important role to facilitate the usage of the system. Hence, we presented the design principle of the architecture of the system, the service enabling layer, and spoken language processing techniques in multi-modal interaction, etc.

Keywords: Multi-modal, Ozone, MIAMM, Dialogue, Spoken Language, XML.

1 Introduction

Nowadays, the quality of human life has been improved by the pervasive network and communication framework which is expected to provide information and services to individuals from anywhere and at anytime. Offering users with an invisible but easy-to-use environment is currently a hot topic in the information and communication technology community. The Ozone project is to meet the increasing requirements in the consumer domain and support the mobile computing in the future. Now, several similar projects have been launched. MIT Oxygen [1], which is part of the USA Expeditions Initiative funded by the Advanced Research Project Agency, is aiming at the research and development of ambient intelligence applications and appliances oriented for professional IT users. MIAMM [2], which is to develop new concepts and techniques in the field of multi-modal dialogues to allow fast and natural access to multimedia databases and information, is supported and organized by Information Society Technologies (IST) [3] of EU, started in 2001.

Ozone [4], which is also supported by IST of EU and started in 2002, intends to implement a generic architecture and framework that will facilitate the use and acceptance of ambient intelligence in the consumer domain. Therefore, the Ozone project aims at development of novel concepts, techniques and tools to provide invisible computing for the domestic and nomadic personal use of information

[*] This work was partially supported by NSFC (No.60572125).

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 640 – 647, 2006.

technology. One of the important concepts of Ozone is that the applications of the advanced technologies should support the user centric retrieval and consumption of information compared to the computer centric approach widely used in current practice. That requires the human-machine interfaces should be as easy and natural as possible for the average users. To improve the acceptability and usability, multi-modal interactions, i.e., speech, gesture, etc., must be supported. In this paper, we will present the overview architecture of Ozone and we will emphasize the design and implementation of the service enabling layer which manages the communication between users and middleware of the software layer. The integration of spoken language into Ozone system is focused on in this paper.

2 Overview Architecture

In the viewpoint of functionality, the architecture of Ozone system can be depicted as three layers, namely, the *Service-Enabling* layer, the *Software-Environment* layer and the *Platform-Architecture* layer. On top of these three Ozone layers, there is a layer of *Applications* and *Services*, which communicate with users and other systems. The *Services* are then divided into application-related services and external services.

2.1 The Service Enabling Layer

The Service-Enabling layer and the Software-Environment layer together form the *middleware* of the Ozone system. The Service-Enabling layer has the characteristics to enable context awareness, multi-modality, and security. The functions in this layer can be classified as context awareness and user interface, etc.

2.1.1 Context Awareness

Management for context awareness is essential in the Ozone system. This includes gathering and combining sensor services and their output, and reasoning about the implications, and making this available to other interested parties, via high-level context services for example, that provide access to the Context Model. Apart from this sensor-based context awareness, awareness of the history of activities, and the occurrence of other concurrent activities may be taken into account. Community Management: for supporting social groups of users, as a basis for sharing and communicating, possible based on the members' contexts. Also, Preference Management (both of end-users but also settings for devices and functionalities) is required. Finally, Profile Management (related to preferences, but more focused on 'what content', than on 'how') is needed. These functions may use the Knowledge-Store service for persistently storing their data.

2.1.2 User Interface

The user interface (UI) enables the natural man-machine interaction between the end-user and the system. It handle device-variety and enable multi-modality. Obviously, speech, vision and touch are very natural ingredients for a multi-modal user-interface. Ideally, we want to decouple applications from the availability of interaction functionalities (like typing versus speech), which may vary over devices, and with time. The UI Management function is the intermediary for this. Depending on the dynamically available interaction functionality-control services, a set of Multi-Modal Widgets is offered to the application. For example, if the application expresses the abstract need for text-input, it may be offered a text-box widget for keyboard input, and a speech-to-text widget for speech input. Obviously, the most natural combination of choices should be offered, making use of the user's context, preferences, profiles, *et cetera*. The latter (choosing) is the task of the Smart Agent.

2.2 The Software Environment Layer

The Software Environment layer has the characteristics to enable seamless operation, interoperability, and extendibility. The functions in this layer can be classified as platform infrastructure, service infrastructure, application and content infrastructure.

2.2.1 Platform Infrastructure

The functions in this class deal with the management of the devices themselves, the relation amongst the devices, the functionalities within the devices, and the network, all via the Ozone-compliant platform interface. All devices are able to boot autonomously, the Booting functions takes care of this. Then the devices discover each other in the network, adding plug-and-play functions, which are taken care of via the Device Discovery and Lookup, to access device-control services of other devices in the network.

Device Management enables upgrading, extending, and patching the device with new hardware and software, throughout the lifetime of the device, via the device-control service. Resource Management deals with sharing functionality-control services and underlying resources, and solving conflicts in simultaneous attempts to use a functionality-control service, when this is not possible. Power Management deals with saving energy, by shutting down (or even shutting off) parts of the device and functionalities when they are not needed, again via the device- and functionality-control services. Network Mobility concerns support for hand-over of connections between multiple networks, allowing for increased mobility of the end-user.

2.2.2 Service Infrastructure

The devices themselves and the functionalities that they contain are encapsulated in device-control and functionality-control services as mentioned before, to make them available to the rest of the Ozone system. In addition, a number of other services are present in the system. This has been explained in the section on Services. The class of Service Infrastructure functions provides the basic mechanisms for all Ozone services to be installed, to discover and to be found, to communicate, *et cetera*. Service Discovery and Lookup allows services to advertise themselves, and other clients (services, applications) to find them. Often this is done via some registry mechanism. For this to work, Services need to be addressable via a name, which is unique within the system. Therefore a Naming scheme for services is required. Also Services need to communicate, and communicated to, which is listed as service-control communication. Furthermore functions for Signaling Events and Transactions are provided. This concerns the interaction between all types of services and applications. Service Composition includes support for composing services based/aggregated from other existing services.

2.2.3 Application and Content Infrastructure

In the Ozone system, there is a decoupling between the reference to content (a unique name, therefore an Content Naming Scheme is required), and the actual instances of the referenced content on the distributed Storage functionalities. This mapping is performed by the Content Discovery and Lookup function. The appropriate or most suitable application is launched via some Application Startup function. Also, the Migration of Applications function supports follow-me applications for the end-user. For Applications to process the content, data streams will be set up from the source service of the content, to the sink services. The Stream Management function checks the possibility for and actually sets up these streams. Also, it includes stream-processing functionality-control services, like trans-coders, if required.

The plug model, that allows for conceptually setting up streams between source services and sink services via some form of plug matching, is provided. Functionality-control services offer functionality plugs that have attributes like encoding (MPEG2) related to the content that will be transported via the connection. Device-control services offers device plugs that have attributes like protocol (http, RTP) related to the network connection for streaming between the devices. The function for the synchronization of content across the distributed storage functionalities is also essential, when replication and caching of content is done.

2.3 The Platform-Architecture Layer

The Platform-Architecture layer consists of the device-platform and the network connecting these devices, and has the characteristics to enable high performance, adaptability and re-configurability. This layer can be divided into two sub-levels: the lower part which is proprietary and exposed via the proprietary device-platform interface, and the higher-part, which makes the platform Ozone intra-device compliant, by abstracting from proprietary implementations through offering the ozone platform interface.

3 The Multi-modal UI

The Service-Enabling layer of Ozone system must communicate with users, applications and the middleware of the software-enabling layer. To make the human-machine interaction easier, the multi-modal UI using speech, vision, touch, etc., is offered in this layer. The detailed structure of the Service-Enabling layer is depicted as Figure 1.

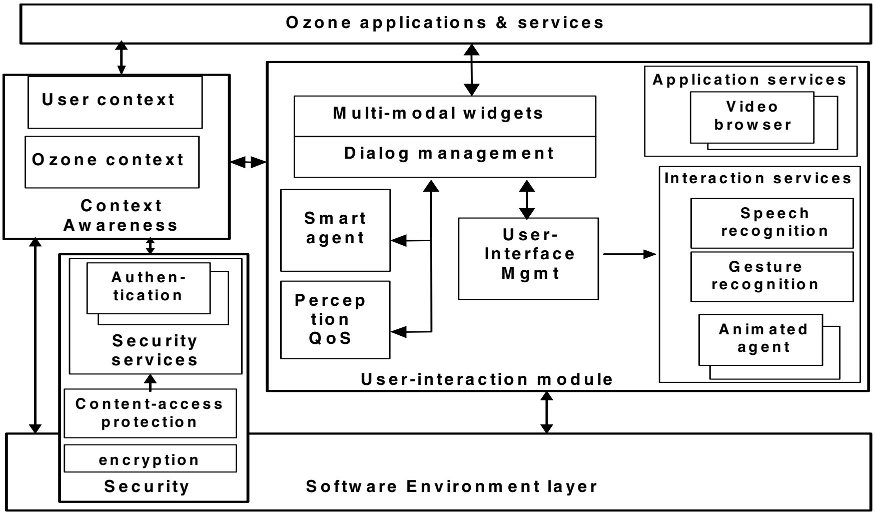


Fig. 1. The structure of the Service-Enabling layer

3.1 Interact with the User

Dialog management, based on the requests of an Ozone application, plans the interactions with the user and uses the UI management to realize this plan. The role of the smart agent is to facilitate the communication between the user and the system, by taking into account the user preferences. Natural interaction with the user replaces the keyboard and windows interface with a more natural interface like speech, vision, touch or gestures.

3.2 Model the User Behavior

The user context stores all the relevant information concerning a user, automatically builds the user preferences from his past interactions and eventually abstracts the user profile to more general community profiles.

3.3 Model the Ozone Context

The Ozone context takes care of the world model in the context of Ozone. This essentially deals with the list of authorized users, available devices, active devices, state of the system, et cetera.

3.4 Control Security Aspects

The security module ensures the privacy and security of the transferred personal data and deals with authorization, key and rights management.

Multi-modality in Ozone is the abstraction of classical user-interface provided by graphical user interfaces. These multi-modal functions are then instantiated in the form of a sequence of actual user interactions by means of speech recognition, gesture interpretation, animated agent, video-browser actions and so forth. The Ozone framework support user interaction combining voice, pointer and keyboard selection. Furthermore, the Ozone framework provides matching output modalities, such as talking and gesturing virtual presenters. In particular, a humanoid modality is available by means of a virtual presenter to communicate understanding, expectations and readiness; a presenter can be personal or service-specific.

4 UI Via Spoken Language

Speech recognition (ASR) and text-to-speech (TTS) techniques provided a natural way to operate machines via spoken language [5,6,7]. Hence speech is used in Ozone as a choice among other UI means. The Ozone framework can offer user-independent speech recognition that is adaptive to the user's specific situation (e.g., accounting for the environment noise as in the car). Dually, the Ozone framework supports speech synthesis by means of talking-head. To integrate the speech I/O action into the multi-modal UI, an XML-based description language is defined to clearly represent the interaction between speech I/O and dialogue manager. The figure 2 shows the communication between speech I/O module and the dialogue manager.

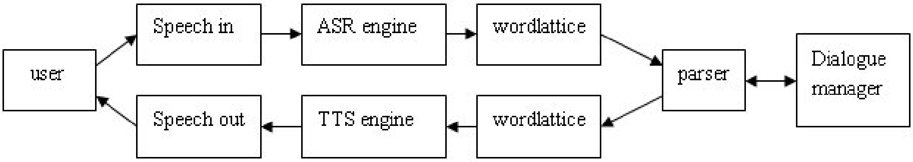


Fig. 2. The UI via spoken language in Ozone

As above figure 2 shown, the speech recognizer takes the raw speech signal as input, and generates a word lattice as output. The word lattice is represented by an XML-based description language and parsed by an action parser. The result (actions) of the parser is then sent to the dialogue to realize the corresponding acts. The figure 3 shows the word lattice generated by the speech recognizer ESPERE.

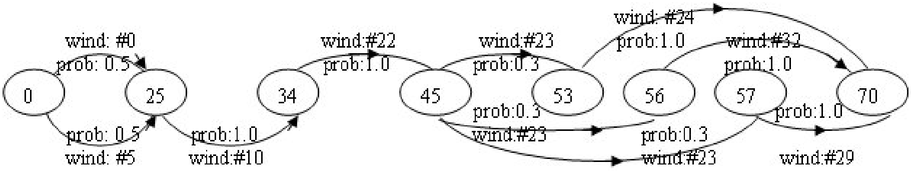


Fig. 3. word lattice decoding of the speech recognizer ESPERE

The word lattice shown in figure 3 is then represented by an XML-based language as the following (not complete):

```
<Block num="0" defaultSpeakerInfoRef="">
   <Node num="0" timeOffset="0">
         <WordLink nodeOffset="25" probability="  0.50" word="0"/>
         <WordLink nodeOffset="25" probability="  0.50" word="5"/>
   </Node>
   <Node num="25" timeOffset="57">
         <WordLink nodeOffset="9" probability="  1.00" word="10"/>
   </Node>
   <Node num="34" timeOffset="122">
         <WordLink nodeOffset="11" probability="  1.00" word="22"/>
   </Node>
   <Node num="45" timeOffset="160">
         <WordLink nodeOffset="8" probability="  0.33" word="23"/>
         <WordLink nodeOffset="11" probability="  0.33" word="23"/>
         <WordLink nodeOffset="12" probability="  0.33" word="23"/>
   </Node>
   <Node num="53" timeOffset="185">
         <WordLink nodeOffset="17" probability="  1.00" word="24"/>
   </Node>
      <Node num="56" timeOffset="207">
         <WordLink nodeOffset="14" probability="  1.00" word="32"/>
   </Node>
   <Node num="57" timeOffset="211">
         <WordLink nodeOffset="13" probability="  1.00" word="29"/>
   </Node>
   <Node num="70" timeOffset="299"/>
</Block>
```

5 Applications

The application based on Ozone architecture can be illustrated as follows. At the bottom, the device internals including five functionalities are depicted: a display functionality that drives an external screen, a speaker functionality that drives external set of speakers, a storage functionality that exposes in internal hard-disk drive, a decompression functionality that implements a (here: MPEG4) decoder purely in software on top of the proprietary RTE, and a network-interface functionality that drives a (here: wired Ethernet) network.

The five device-related functionalities are exposed to the rest of the Ozone system as (in this case) two functionality-control services: an AV-render service that wraps the proprietary display plus speaker plus decompression functionality, and a content-store service that wraps the proprietary storage functionality, as indicated by the one-way arrows. The Registry service is a middleware-related service that implements some form of service discovery and lookup. As an example of an application-related service, a videoconference service is depicted, together with the videoconference application on top. This videoconference service may serve as the meeting point for the multiple videoconference applications on the devices of the multiple end-user taking part in the videoconference.

6 Summary

We discussed the architecture and key issues in the implementation of the Ozone project which is a complicated system using multi-modality to facilitate human beings to use electric machines. In fact, there are still some issues ignored in this paper, such as gesture recognition, user profile, animation simulation, etc. We hope this paper can illustrate the state-of-the-art technology in the human-machine interaction by multi-modality.

References

[1] MIT Oxygen project, http://oxygen.lcs.mit.edu/, 2000

[2] MIAMM project, http://www.loria.fr/projets/MIAMM/, 2002

[3] IST homepage, http://www.cordis.lu/ist/, 2003

[4] Ozone project, http://www.ics.ele.tue.nl/~mgeilen/ozone/, 2003

[5] L. R. Rabiner and B. H. Juang, Fundamentals of Speech Recognition, Prentice-Hall International, Inc. 1993

[6] Kai-fu LEE, Hsiao-wuen HOU and R. Reddy, An overview of the SPHINX speech recognition system, IEEE Transactions on ASSP, January, 1990

[7] X. D. Huang, A. Acero, H. Hon, and S. Meredith, Spoken Language Processing, Prentice Hall, Inc., 2000

Spoken Correction for Chinese Text Entry

Bo-June Paul Hsu and James Glass

MIT Computer Science and Artificial Intelligence Laboratory
32 Vassar Street, Cambridge, MA 02139, USA
{bohsu, glass}@mit.edu

Abstract. With an average of 17 Chinese characters per phonetic syllable, correcting conversion errors with current phonetic input method editors (IMEs) is often painstaking and time consuming. We explore the application of spoken character description as a correction interface for Chinese text entry, in part motivated by the common practice of describing Chinese characters in names for self-introductions. In this work, we analyze typical character descriptions, extend a commercial IME with a spoken correction interface, and evaluate the resulting system in a user study. Preliminary results suggest that although correcting IME conversion errors with spoken character descriptions may not be more effective than traditional techniques for everyone, nearly all users see the potential benefit of such a system and would recommend it to friends.

1 Introduction

The number of Chinese-speaking Internet users has quadrupled over the past 5 years to over 132 million today [1]. With more than 120 million instant messaging users and 60 million bloggers projected in China alone by the end of 2006, efficient Chinese text entry plays an ever increasing role in improving the overall user experience for Chinese speakers [2,3].

Unlike text entry in English, the individual keys on the standard keyboard do not map directly to Chinese characters. Instead, an input method editor (IME) transcribes a sequence of keystrokes into characters that best satisfy the specified constraints. Phonetic IMEs are a popular category of Chinese IMEs that interpret the keystrokes as the pronunciations of the input characters. However, in Traditional Chinese, more than a dozen homonym characters commonly share a single pronunciation. Thus, the IME often leverages a language model (LM) to select the character sequence that maximizes the sentence likelihood [4].

The process of converting phonetic input into the corresponding characters is known as pinyin-to-character, phoneme-to-character, or syllable-to-character conversion [4,5,6]. Popular phonetic alphabets include zhuyin (注音), also known as bopomofo (ㄅㄆㄇㄈ), and pinyin (拼音). Recent advances in phoneme-to-character conversion have improved the character conversion accuracy to above 95% on newspaper articles [6]. However, the accuracy is reduced on text with mismatched writing styles and is significantly lower on out-of-vocabulary words in the LM. Consequently, efficient text entry requires an effective correction mechanism for users to change the incorrect homonyms to the desired characters.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 648–659, 2006.

The Microsoft New Phonetic IME (MSIME) (微軟新注音輸入法) [7] is a popular IME for Traditional Chinese input. To correct a conversion error when using the MSIME, the user first moves the cursor to the incorrect character and then selects the desired character from a candidate list of homonyms with matching pronunciations, as illustrated in Fig. 1. For errors far from the current cursor position, navigating to the target position can be tedious. Since some pronunciations have more than 200 matching characters, the candidate list is often divided into multiple pages. While the desired character often appears within the first page and can be selected with a single keystroke, visually finding the correct character can at times be painstaking given that characters are rendered with a small font and sometimes differ only by their radicals.

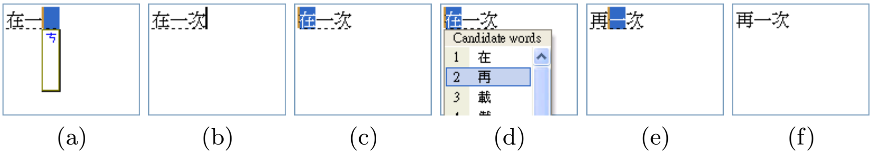


Fig. 1. Illustration of steps involved in correcting the character 在 in 在一次 to 再. After the phonetic sequence is entered in zhuyin (a,b), the user first highlights the conversion error (c). Next, the user selects the desired character from the drop-down candidate list (d) and commits the correction (e). Once all characters in the IME composition window have been corrected, the user commits the composition (f).

In an e-mail survey conducted with 50 Chinese typists, 40% reported skipping past the target character accidentally more than 5% of the time when scanning the candidate list. Due to the frustrating nature of current correction interfaces, 56% admitted that they sometimes do not correct conversion errors, especially in informal text conversations with close friends. With intelligent IMEs that learn from the words and phrases entered by the user [7,8], leaving conversion errors uncorrected further reinforces the errors and increases the likelihood of the system making similar errors in the future.

In this work, we explore the use of a novel spoken correction approach to address some of the shortcomings in current correction interfaces. Specifically, leveraging users' familiarity with describing the characters in their names when making self-introductions, we support spoken correction via usage, structure, radical, or semantics description of the desired character. For example, to correct the IME composition 在一次, we can say the phrase 再見的再 to specify the desired character 再 from its usage 再見.

In the following sections, we first provide additional background on Chinese text entry and discuss related work. Next, we compute various statistics involving Chinese homonyms and analyze how users disambiguate among them using character descriptions. We then describe the design and implementation of the spoken correction interface and evaluate the system through a user study. Finally, we discuss observations from the user study and areas for future work.

2 Background

2.1 Chinese Text Entry

Popular IMEs for Chinese text entry generally can be categorized as editors that input characters by either compositional structure or pronunciation. While IMEs based on character structures, such as Changjie (倉頡), Boshiamy (嘸蝦米), and Wubi (五筆), often allow for fast entry rates with infrequent conversion errors, they typically require users to learn a set of decomposition rules that take time to master. On the other hand, phonetic IMEs using phonetic alphabets, such as New Phonetic (新注音) and Natural (自然), require minimal learning for most users, as they are taught phonetic spelling in school. Although phonetic methods generally do not involve more keystrokes than structural methods initially, it incurs more conversion errors due to the large number of homonyms per syllable pronunciation. With each correction requiring a visual search for the desired character and additional keystroke for navigation and target character selection, the correction of even a small percentage of characters can account for a significant portion of the overall entry time. Thus, the overall character entry rate of experienced users of phonetic IMEs is typically lower than those using structural input.

2.2 Related Work

Tsai et al. [9] applied spoken descriptions of characters to help resolve homonym ambiguities in Chinese names for a directory assistance application. In addition to generating character usage descriptions from automatically extracted words, phrases, and names, a list of character descriptions for the most common last names was manually collected. With 60,000 descriptions for 4,615 characters, the character description recognizer achieved a success rate of 54.6% at identifying the target character.

In this work, we apply the approach of using character descriptions for disambiguating among homonyms as a correction interface for Chinese text entry using IMEs. We observe that in addition to describing characters by usage phrase (e.g. 再見的再), descriptions using character radical (女字旁的她), compositional structure (土川圳), and character semantics (女生的她) are also fairly typical. In addition, since these descriptions include the target character at the end, the position of the desired character within the current IME composition can often be unambiguously inferred from the character description. Furthermore, because the pronunciations of the characters in the uncommitted IME composition are known, we can limit the recognizer grammar to only accept descriptions for characters with those pronunciations, reducing the grammar perplexity.

Leveraging these observations, we have extended the commercial MSIME with the capability for users to correct errors in the conversion using spoken character descriptions. Preliminary results from user studies suggest that with additional refinements and improvements to recognition accuracy, spoken correction using character descriptions has the potential to improve the correction experience for a significant group of Chinese typists.

3 Analysis

3.1 Homonym Statistics

Due to the obscurity of many characters and the continuous introduction of new characters, the number of Chinese characters varies significantly depending on the particular dictionary or computer character encoding. The CNS11643 standard, for example, defines over 48,000 characters, although many are unpronounceable and the average person only uses around 5,000 characters [10]. The distinction between traditional (繁體) and simplified (簡體) Chinese introduces further complications as many character sets include characters from both styles. In this work, we will only consider the set of characters that can be phonetically entered via the MSIME.

We gathered two text corpora for frequency analysis and system evaluation. The first corpus, *CNA2000*, consists of newswire articles from the Central News Agency of Taiwan in the year 2000 [11]. Specifically, we considered only the headline and core news content for the analysis. For the second corpus, *Blogs*, we extracted text excerpts from 10,000 RSS feeds of randomly selected blogs from a popular blogging website in Taiwan. For both corpora, we segmented the content at punctuations, symbols, and other non-Chinese characters and discarded segments containing character outside our character set. Although blogs better match the informal style of most text entry scenarios, they are also more likely to contain conversion errors that the writer neglected to correct. For simplicity, we will treat both corpora as containing the correct reference text.

To gain insight into the homonym problem in Chinese, we computed, in Table 1, various statistics relating characters to their pronunciations, specified with and without tone. Although there are only 16.8 characters per pronunciation on average, the number of homonym characters with the same pronunciation averaged over the character set is over 38. In the worst but not infrequent case, the candidate list for the pinyin *yi4* has over 207 items. Fortunately, through the application of the language model to order the characters in the candidate list, more than 96% and 95% of the target characters appear on the first page, when correcting conversion errors in a simulated entry of the text from a random subset of the *CNA2000* and *Blogs* datasets, respectively.

Table 1. Statistics on the pronunciations of the 19,991 characters in the character set

(average / max)	With Tone	Without Tone
# Pronunciations	1387	408
# Characters per Pronunciation	16.8 / 207	54.4 / 383
# Homonyms per Character	38.2 / 206	101.4 / 382
Average Rank of Target Character		
CNA2000	3.0	6.4
Blogs	3.0	6.2

3.2 Character Description

To better understand how character descriptions disambiguate among homonym characters, we asked 30 people to describe the characters in their Chinese name. In a separate study with 10 participants, we requested descriptions for 50 randomly selected characters from among the 250 most frequently confused characters by the IME, displayed next to the incorrect homonyms. Most of the 587 character descriptions collected can be classified into one of the description types listed in Table 2, where we also provided analogous English examples.

Table 2. Types of character descriptions with typical templates, Chinese examples, and approximately analogous examples in English. The target character is in bold.

Description	Typical Template	Example	Approximate English Analogy
Usage	[*usage phrase*]的[*char*]	希望的**希**	**lead** as in lead paint
Structure	[*composition*][*char*]	人白**伯**	**rainbow**, rain plus bow
Radical	[*radical name*]的[*char*]	草字頭的**蔡**	**dialog** with the Greek root log
Semantics	[*meaning*]的[*char*]	數字的一	**red** as in the color
Strokes	Character-dependent	三横一豎王	**H** with 2 vertical and 1 horizontal strokes
Compound	Speaker-dependent	微笑的**微**	psych as in psychology
	[*char*] usually omitted	加草字頭 (**薇**)	with an extra E at the end (**psyche**)

When describing by usage, the description is generally a word phrase, idiom, or proper name, consistently in the form [*usage phrase*] 的('s) [*target character*]. While most structural descriptions specify the character by its subcomponents, a few users describe some characters by removing components from more easily describable characters. For example, the character 念 can be described as 唸書的唸, 沒有口字旁. Furthermore, when the desired character differs from the incorrect character by a single component, it is often natural to base the description on the current character. Thus, to change 啊 to 阿, one might say 沒有口的啊 (啊 without 口).

Character descriptions by radical generally can be derived from the radical name and a few simple templates. However, some of the 214 radicals have common aliases, especially when appearing in an alternate form or in a particular position within the character. For example, both 拿 (take) and 打 (hit) share the radical 手 (hand), which can be described using the standard template 手部的[拿,打]. However, because the radical 手 appears in an alternate form in the character 打, 提手旁的打 is another popular description for 打.

Some characters, such as 她 (she) and 九 (nine) are most commonly associated with their semantics, rather than their usages, structure, or radical. For these, special character-dependent descriptions are often used, such as 女生的她 (female's she) and 數字的九 (number's nine). Although descriptions using strokes are also character-dependent, they are specific and do not vary across speakers.

In Table 3, we summarize the observed occurrences of each description type for characters from last names (Last), first names (First), and the most frequently confused characters (Confused). Overall, descriptions using usage dominate all other description types, except when describing last names. Since the characters in last names differ significantly in distribution from the characters in first names [12], it is not surprising that their description type distributions are also different. However, in addition to the dependency on the specific character, character descriptions also depend on the context. For example, whereas most people would describe the last name 許 by its structure 言午許, in the context of a sentence, many would describe the same character by its usage 許多的許 instead.

Table 3. Occurrences of each character description type from user studies

Description	Last	First	Confused
Usage	8	53	400
Structure	17	1	8
Radical	3	1	45
Semantics	0	1	25
Strokes	2	1	0
Compound	1	0	2
Others	0	0	19

To measure the variability across speakers in the descriptions of a character, we computed the normalized entropy of the character descriptions for each character spoken by at least 5 participants. For a sample size of N, we define the normalized entropy H_0 as the entropy of the empirical distribution divided by $\ln(N)$. Thus, if all samples have the same value, $H_0 = 0$. If each sample has a different value, $H_0 = 1$. As shown in Table 4, the normalized entropy for most characters are significantly less than 1. Although each character can be described in numerous ways, only a few descriptions are commonly used across users in general. Thus, an effective spoken correction system should not only accommodate the different description types, but also leverage the limited variability of character descriptions across users to improve the speech recognition accuracy.

Table 4. Normalized entropy of character descriptions. For example, of the 7 description instances for the character 集, there are 6 集合的集 and 1 集中的集. Thus, the normalized entropy is $H_0 = -\left(\frac{6}{7}\log\frac{6}{7} + \frac{1}{7}\log\frac{1}{7}\right) / \log 7 = 0.21$.

Norm. Ent.	# Chars	Example
0.0–0.2	9	雨(0.00): 下雨的雨 7
0.2–0.4	7	集(0.21): 集合的集 6, 集中的集 1
0.4–0.6	19	一(0.41): 一二三四的一 5, 一二三的一 4, 一個人的一 1
0.6–0.8	9	不(0.66): 不是的不 3, 不要的不 2, 不可能的不 1, 不好的不 1
0.8–1.0	7	來(0.83): 來去的來 2, 來了的來 1, 來往的來 1, 起來的來 1

4 Design

To investigate the use of spoken character description as a correction interface for Chinese text entry, we extended the MSIME with the capability to correct homonym errors using the usage, structure, radical, or semantics description of the desired character. With the Microsoft Speech API 5.1, we built a custom context-free grammar (CFG) for the spoken character descriptions and used the Microsoft Chinese (Traditional) v6.1 Recognizer as the speech recognizer [13,14]. The following sections describe the construction of the character description grammar and the design of the correction user interface in more detail.

4.1 Grammar Construction

As observed in Sect. 3.2, character descriptions by usage, compositional structure, and radical generally follow specific templates, allowing for automatic generation. In the user study, the few users who initially deviated from the typical templates showed no difficulty adjusting after being instructed on the expected patterns. Unfortunately, character descriptions by semantics and strokes cannot be generated automatically and required manual data collection. Thus, given the constrained descriptions, we chose to build a language model consisting of a finite state network of data-driven and manually collected character descriptions.

To build usage descriptions, we extracted all word phrases with 2 to 4 characters from the CEDict Chinese-English Dictionary [15], for a total of 23,784 words (辭), idiomatic phrases (成語), and proper names (專有名詞). For each character in each word phrase, we added to the grammar a usage description of the form [*word phrase*]的[*char*].

The Chinese Character Structure Database (漢字構形資料庫) provides the structure information for 7,773 characters in the IME character set [16]. From this, we added simple compositional descriptions of the form [*composition*][*char*]. We leave support for more complex structural descriptions to future work.

Most radicals can be described with a few template expressions. For example, the radical 人 may be described using 人部, 人字部, 人字旁, or 人字邊. However, some radicals also have additional aliases, such as 單人旁 for the radical 人. Thus, to build character descriptions using radicals, we manually identified a set of template expressions appropriate for each radical and supplemented it with a list of radical aliases obtained from the Table of Chinese Radical Names (漢字偏旁名稱表) [17]. Finally, for each character in the IME character set and each corresponding radical name, we added character descriptions of the form [*radical name*]的[*char*] to the grammar.

A single IME composition generally contains only a small subset of the 1,387 pinyin pronunciations. Since users only need to disambiguate among characters whose pronunciation appears within this subset, it suffices to dynamically constrain the language model to only those character descriptions. Thus, when building the CFG, we grouped the character descriptions by the pronunciation of the target character and built a separate rule for each pronunciation. Depending on the IME composition, we selectively activated the appropriate grammar rules to improve both recognition speed and accuracy.

To further speedup the recognition and reduce the grammar size, we optimized the finite state network by merging all character arcs with the same pronunciation, in effect determining the network at the syllable level. To recover the target character, we encoded it in the grammar as a property tag.

However, when reduced to syllables, not all character descriptions yield unique characters. For example, the phrase 就是的就 actually shares the same phonetic representation as 救世的救 and 舊式的舊. In Table 5, we summarize the statistics on the number of character descriptions with identical pronunciations. Although radicals are often the easiest to describe, they are also the most ambiguous on average. Given the ambiguities associated with even character descriptions, an effective correction user interface will need special handling for this condition.

Table 5. Statistics on character descriptions with the same pronunciation

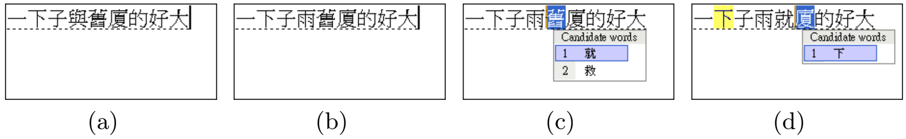
(avg / max)	# Descriptions / Pron		Example
	With Tone	Without Tone	
Usage	1.03 / 4	1.11 / 21	就是/救世/舊式 (*jiu4 shi4*)
Structure	1.04 / 6	1.15 / 8	盻/栖/桸/睎/榽/樨 (*xi1*)
Radical	1.34 / 10	1.82 / 16	泄/洩/浥/液/溢/泆/泆/潩/澺/瀷 (*yi4*)

4.2 User Interface Design

To enable transparent switching between the traditional and the new spoken correction interfaces, we reassigned the `Control` key while composing text with the IME to act as a push-to-talk microphone button for the character description speech recognizer. For each correction, the user may choose to select the target character using the arrow keys as before or press the `Control` key to speak a character description. To simplify end-point detection in the initial implementation, we require the microphone button to be depressed while talking.

After the microphone button is depressed, we enable the grammar rules corresponding to user-specified pronunciations in the current IME composition and begin listening for a character description. Upon a successful recognition, we look up the potentially multiple candidate characters matching the description. Typically, the recognized target pronunciation only corresponds to a single syllable position in the IME composition. Thus, if the character description specifies a unique candidate, we immediately replace the character at the matching position with the user-described character, as illustrated in Fig. 2(b). If the character description matches multiple characters, a list containing the candidate characters is displayed at the matching syllable position, as shown in Fig. 2(c). Ideally, the list will be sorted by the language model likelihood. As an approximation, we sort the list according to the ordering of these characters in the original IME candidate list. Since users are unlikely to describe the currently hypothesized character, it is explicitly moved to the bottom of the list, if included.

Occasionally, the pronunciation corresponds to multiple candidate syllable positions, requiring user intervention prior to making the correction. To allow the user to select the syllable from among these candidate positions, we highlight all candidate positions, display the filtered candidate list containing the matching characters, and restrict the left/right arrow keys to navigate only among these positions, as illustrated in Fig. 2(d). To reduce keystrokes, the candidate list is initially displayed under the position corresponding to the single correction that maximizes the language model likelihood.



(a) (b) (c) (d)

Fig. 2. Illustration of the steps involved in correcting conversion errors. Entering the phonetic sequence for 一下子雨就下的好大 yields the IME hypothesis 一下子與舊廈的好大 (a). The user depresses the microphone button and says 下雨的雨 to correct 與 with 雨. Since this character description uniquely identifies the character 雨 and only corresponds to a single position, the system automatically replaces the error 與 with 雨 (b). To describe 就, the user speaks the usage phrase 就是的就. In this case, because 救世的救 is acoustically identical to this character description, the system shows the candidate list to allow the user to specify the desired character (c). Finally, the user says 下面的下 to replace 廈 with 下. Because two positions in the IME composition contain the syllable *xia4*, the system highlights both candidate positions and selects the one most likely to contain the error (d). In this case, the candidate list appears under 廈 since the first position already contains the specified character 下.

5 User Study

For evaluation, we conducted a user study with 10 students from Taiwan with varying proficiency in Chinese text entry. The study included a questionnaire on the participant's experience with Chinese input, approximately 5 minutes of speech recognition enrollment for acoustic model adaptation, and a collection of 50 spoken character descriptions. Participants were also asked to enter 2 distinct sets of 20 Chinese sentence fragments with the IME, one using traditional keyboard correction, the other using spoken correction with character descriptions. Sentences from both sets were manually selected from the *Blogs* corpus to contain one or more conversion errors. The two sets were randomly alternated for each participant to remove any bias resulting from differences between the two sets.

Table 6 summarizes the results from the study. Overall, the response to spoken correction is positive, with half of the participants expressing interest in using the system. Through the post-study questionnaire, we learned that of the 5 users expressing a neutral or negative opinion, 3 have memorized deterministic key sequences of common characters for their respective IMEs. Thus, minor improvements to correcting the sporadic errors that they encounter do not justify overcoming the learning curve of a new system and the need to set up a high-quality microphone whenever performing text entry. Interestingly, of these 5 users without definite interest in using the system themselves, 4 would still recommend it to friends. As user J observed, "This system is very useful and convenient for users less familiar with Chinese input... [However], frequent typists will still choose selecting characters [using the keyboard]." Thus, although spoken correction may not be more effective for everyone, nearly all participants saw the potential value of such a system, even with less than 10 minutes of usage.

Table 6. Summary of user study results. Prior to the survey, we asked participants to estimate the average amount of time per week they spend entering Chinese text and indicate the IME they use most frequently. After the study, in which the user had a chance to enter text using both the traditional keyboard correction and spoken correction, we asked users if they would consider using spoken correction in the future and recommend the system to a friend.

User	A	B	C	D	E	F	G	H	I	J
Use Spoken Correction	Y	Y	Y	Y	Y	M	M	M	M	N
Recommend to Friends	Y	Y	Y	Y	M	M	Y	Y	Y	Y
Usage/Week (hr)	1	2	2	2	7	1	3	4	6	2
Typical IME	NP	NP	NP	N	NP	CJ	HI	P	G	NP

NP: New Phonetic 新注音 P: Phonetic (舊)注音 CJ: Changjie 倉頡
N: Natural 自然 HI: Hanin 漢音 (Mac)
Y: Yes N: No M: Maybe

6 Discussions

One concern with spoken correction is the cognitive load associated with identifying an appropriate description for the target character. Unlike the characters in their names, all users experienced some degree of difficulty describing certain characters, such as 之 (possessive particle), that are not associated with common word phrases and are difficult to describe by radical or structure. However, once a description for a difficult character is suggested, the participants did not encounter any difficulty recalling the description the next time the character is observed a few minutes later.

Many factors contribute to the difficulty of describing characters. As observed in Sect. 3.2, users naturally describe characters by usage in a word phrase. However, this may not always be the most effective approach. Although less natural, it is sometimes easier to identify a character by its compositional structure or radical. For characters from a word phrase in the target sentence, many users have the false notion that because the IME converted the character incorrectly, using the same word phrase to describe the character will not fix the error. As analyzed in [6], more than a third of conversion errors from a bigram-based IME are due to segmentation errors. Thus, explicitly specifying the segmentation boundary through character descriptions can actually correct many of these errors.

Lastly, in the current design, users generally cannot attribute the cause of misrecognitions to acoustic mismatch or unexpected character description, as they have identical behavior. Using the preliminary grammar constructed for the user study, out-of-grammar character descriptions account for 35% of the total spoken corrections. Of the in-grammar descriptions, 16% contained recognition errors. Thus, to improve the spoken correction system, we need to not only improve the grammar coverage, but also mitigate the effect of recognition errors.

7 Conclusion and Future Work

In this work, we introduced a novel correction interface for Chinese text entry using spoken character descriptions. Specifically, we identified common approaches people use to describe characters and constructed an automatically generated character description grammar from various lexical corpora. Finally, we evaluated a preliminary implementation of the spoken correction interface system through a user study that demonstrates the potential benefit of the spoken interface to a considerable subset of Chinese typists.

As shown in Sect. 3.2, most users describe characters using a small subset from among all potential descriptions. Thus, an effective approach to improving the recognition accuracy is to weigh the different character descriptions by their likelihood of utilization. Furthermore, as observed with difficult-to-describe characters, once users identify a successful description for a character, they tend to reuse the same description again for future instances of the character. This suggests that we can further improve the language model performance by emphasizing previously observed character descriptions.

For future work, in addition to incorporating more data to improve grammar coverage, we would like to explore such language model adaptation techniques to reduce the recognition error rate. We also hope to incorporate various feedback from the user study participants to improve the user interface design. Finally, to reduce the effort in evaluating changes to the system, we plan to simulate user input and measure the overall system performance.

In this paper, we focused on applying spoken character descriptions to Chinese keyboard IMEs. However, the approach generalizes to other East Asian languages, such as Japanese and Korean and even to text entry via handwriting and speech, where there are ambiguities in the resulting text. With the rapid growth in text input on mobile devices, we would also like to study the application of spoken correction to text entry interfaces using the keypad.

Acknowledgement. We would like to thank Ingrid Bau for the numerous design validation discussions, scheduling of the user studies, and assistance with data transcription. Further credit goes to Chao Wang for the helpful discussions, feedback on the user study design, and detailed review of the paper drafts. The speech recognition software used in this work is provided by Microsoft Corp.

References

1. Miniwatts Marketing Group: Internet users by languages. Internet World Statistics Website (Mar. 31, 2006) http://www.internetworldstats.com/stats7.htm
2. iResearch Inc.: The Number of IM Users Will Reach 200 million by 2010 in China. iResearch–China Internet Research Center Website (May 17, 2006) http://english.iresearch.com.cn/instant_messenger/detail_news.asp?id=6938
3. Xinhua News Agency: Chinese bloggers to reach 100 million in 2007. China View Website (May 6, 2006) http://news.xinhuanet.com/english/2006-05/06/content_4513589.htm
4. Gao, J., Goodman, J., Li, M., Lee, K.: Toward a Unified Approach to Statistical Language Modeling for Chinese. ACM Transactions on Asian Language Information Processing, Vol. 1, No. 1 (2002) 3–33
5. Hsu, W., Chen, Y.: On Phoneme-To-Character Conversion Systems in Chinese Processing. Journal of Chinese Institute of Engineers 5 (1999) 573–579
6. Tsai, J., Chiang, T., Hsu, W.: Applying Meaningful Word-Pair Identifier to the Chinese Syllable-to-Word Conversion Problem. In Proc. ROCLING (2004)
7. Microsoft Corp.: 微軟新注音輸入法 2003. Microsoft Website (Jun. 15, 2006) http://www.microsoft.com/taiwan/windowsxp/ime/windowsxp.htm
8. IQ Technology Inc.: Natural Chinese Input 8. IQ Technology Website (Jun. 15, 2006) http://www.iq-t.com/en/PRODUCTS/going_01.asp
9. Tsai, C., Wang, N., Huang, P., Shen, J.: Open Vocabulary Chinese Name Recognition with the Help of Character Description and Syllable Spelling Recognition. In Proc. ICASSP (2005)
10. CNS11643 國家中文標準交換碼. CNS11643中文全字庫 Website (Jun. 15, 2006) http://www.cns11643.gov.tw/web/word.jsp#cns11643
11. Graff, D., Chen, K.: Chinese Gigaword. Linguistic Data Consortium (2003)
12. Tsai, C.: Common Chinese Names. Chih-Hao Tsai's Technology Page Website (Dec. 5, 2005) http://technology.chtsai.org/namefreq/
13. Microsoft Corp.: SAPI 5.1. Microsoft Website (Mar. 3, 2003) http://www.microsoft.com/speech/download/old/sapi5.asp
14. Microsoft Corp.: Install and Train Speech Recognition. Microsoft Office Online Website (Jun. 15, 2006) http://office.microsoft.com/en-us/assistance/HP030844541033.aspx
15. Peterson, E.: CEDICT: Chinese-English Dictionary. On-line Chinese Tools Website (Jun. 15, 2006) http://www.mandarintools.com/cedict.html
16. Institute of Linguistics, Academia Sinica: 漢字構形資料庫. 中研院資訊所 Website (Aug. 15, 2005) http://ckip.iis.sinica.edu.tw/CKIP/tool/toolreg_intro.html#HANZI
17. 漢字偏旁名稱表. 楓雪軒 Website (Nov. 17, 2004) http://www.fxx520.com/Article/ShowArticle.asp?ArticleID=365

Extractive Chinese Spoken Document Summarization Using Probabilistic Ranking Models

Yi-Ting Chen[1], Suhan Yu[1], Hsin-Min Wang[2], and Berlin Chen[1]

[1] National Taiwan Normal University, Taipei
[2] Academia Sinica, Taipei
g93470070@csie.ntnu.edu.tw

Abstract. The purpose of extractive summarization is to automatically select indicative sentences, passages, or paragraphs from an original document according to a certain target summarization ratio, and then sequence them to form a concise summary. In this paper, in contrast to conventional approaches, our objective is to deal with the extractive summarization problem under a probabilistic modeling framework. We investigate the use of the hidden Markov model (HMM) for spoken document summarization, in which each sentence of a spoken document is treated as an HMM for generating the document, and the sentences are ranked and selected according to their likelihoods. In addition, the relevance model (RM) of each sentence, estimated from a contemporary text collection, is integrated with the HMM model to improve the representation of the sentence model. The experiments were performed on Chinese broadcast news compiled in Taiwan. The proposed approach achieves noticeable performance gains over conventional summarization approaches.

Keywords: hidden Markov model, probabilistic ranking, relevance model, speech recognition, spoken document summarization.

1 Introduction

Due to the ever-increasing storage capability and processing power of computers, vast amounts of multimedia content are now available to the public. Clearly, speech is one of the most important sources of information about multimedia content, such as radio broadcasts, television programs, and lecture recordings, as it provides insight into the content. Therefore, multimedia access based on associated spoken documents has received a great deal of attention in recent years [1]. However, unlike text documents, which are structured with titles and paragraphs and are thus easier to retrieve and browse, associated spoken documents of multimedia content are only presented with video or audio signals; hence, they are difficult to browse from beginning to end. Even though spoken documents are automatically transcribed into words, incorrect information (resulting from recognition errors and inaccurate sentence or paragraph boundaries) and redundant information (generated by disfluencies, fillers, and repetitions) prevent them from being accessed easily. Spoken document summarization, which attempts to distill important information and remove redundant

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 660 – 671, 2006.

and incorrect content from spoken documents, can help users review spoken documents efficiently and understand associated topics quickly [2].

Although research into automatic summarization of text documents dates back to the early 1950s, for nearly four decades, research work has suffered from a lack of funding. However, the development of the World Wide Web led to a renaissance of the field and summarization was subsequently extended to cover a wider range of tasks, including multi-document, multi-lingual, and multi-media summarization [3]. Generally, summarization can be either extractive or abstractive. Extractive summarization selects indicative sentences, passages, or paragraphs from an original document according to a target summarization ratio and sequences them to form a summary. Abstractive summarization, on the other hand, produces a concise abstract of a certain length that reflects the key concepts of the document. The latter is more difficult to achieve, thus recent research has focused on the former. For example, the vector space model (VSM), which was originally developed for ad-hoc information retrieval (IR), can be used to represent each sentence of a document, or the whole document, in vector form. In this approach, each dimension specifies the weighted statistics associated with an indexing term (or word) in the sentence or document. The sentences with the highest relevance scores (usually calculated as the cosine measure of two vectors) to the whole document are included in the summary. To summarize more important and different concepts in a document, the indexing terms in the sentence with the highest relevance score are removed from the document and the document vector is reconstructed accordingly. Then, based on the new document vector, the next sentence is selected, and so on [4]. The latent semantic analysis (LSA) model for IR can also be used to represent each sentence of a document as a vector in the latent semantic space of the document, which is constructed by performing singular value decomposition (SVD) on the "term-sentence" matrix of the document. The right singular vectors with larger singular values represent the dimensions of the more important latent semantic concepts in the document. Therefore, the sentences with the largest index values in each of the top L right singular vectors are included in the summary [4]. In another example, each sentence in a document, represented as a sequence of terms, is given a significance score, which is evaluated using a weighted combination of statistical and linguistic measures. Sentences are then selected according to their significance scores [5]. In the above cases, if a higher compression ratio is required, the selected sentences can be further condensed by removing some less important terms. A survey of the above extractive summarization approaches and other IR-related tasks in spoken document understanding and organization can be found in [1].

The above approaches can be applied to both text and spoken documents. However, spoken documents present additional difficulties, such as recognition errors, problems with spontaneous speech, and the lack of correct sentence or paragraph boundaries. To avoid redundant or incorrect content when selecting important and correct information, multiple recognition hypotheses, confidence scores, language model scores, and other grammatical knowledge have been utilized [2, 6]. In addition, prosodic features (e.g., intonation, pitch, energy, and pause duration) can be used as important clues for summarization; however, reliable and efficient ways of using these prosodic features are still under active research [7, 8]. Summaries of spoken documents can be presented in either text or speech format. The former has the advantage of easier browsing and further processing, but it is subject to speech recognition errors, as well as the loss of the speaker's emotional/prosodic information, which can only be conveyed by speech signals.

In contrast to conventional approaches, we address the issue of extractive summarization under a probabilistic modeling framework. We investigate the use of the hidden Markov model (HMM) [9] for spoken document summarization, whereby each sentence of a spoken document to be summarized is treated as an HMM for generating the document, and the sentences are ranked and selected according to their likelihoods. In addition, the relevance model (RM) [10, 11] of each sentence, estimated from a contemporary text collection, is integrated with the HMM model for better representation of the sentence model. The experiments were performed on Chinese broadcast news compiled in Taiwan.

The remainder of the paper is organized as follows. Section 2 explains the structural characteristics of the hidden Markov model and the relevance model used in this paper. Section 3 presents the experiment setup and the evaluation metric used for spoken document summarization. The results of a series of summarization experiments are discussed in Section 4. Finally, in Section 5, we present our conclusions.

2 Proposed Summarization Models

2.1 Hidden Markov Model (HMM)

In an ad-hoc IR task, the relevance measure of a query Q and a document D_i can be expressed as $P(D_i \mid Q)$; i.e., the probability that the document D_i is relevant given that the query Q was posed. Based on Bayes' rule and some assumptions, the relevance measure can be approximated by $P(Q \mid D_i)$. That is, in practice, the documents are ranked according to $P(Q \mid D_i)$. Each document D_i can be interpreted as a hidden Markov model (HMM) composed of a mixture of n-gram probability distributions for observing a query Q [9]. Meanwhile, the query Q is considered as observations, expressed as a sequence of indexing terms (or words, or syllables), $Q = w_1 w_2 ... w_j ... w_J$, where w_j is the j-th term in Q and J is the length of the query, as illustrated in Fig. 1. The n-gram distributions for the term w_j, for example the document unigram and bigram models, $P(w_j \mid D_i)$ and $P(w_j \mid w_{j-1}, D_i)$, are estimated directly from the document D_i and linearly interpolated with the collection's unigram and bigram models, $P(w_j \mid C)$ and $P(w_j \mid w_{j-1}, C)$, estimated from a large text collection C. Then, the relevance score of a document D_i and a query Q is calculated by

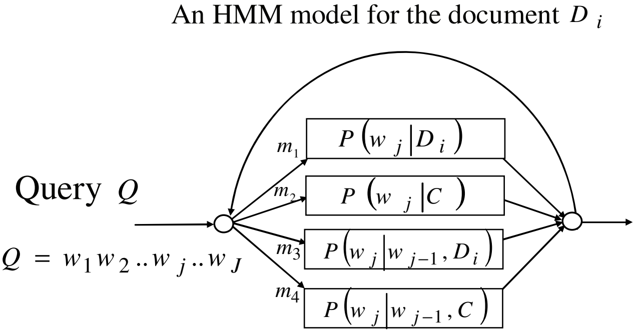


Fig. 1. An illustration of the HMM-based retrieval model

$$P(Q|D_i)_{HMM} = [m_1 \cdot P(w_1|D_i) + m_2 \cdot P(w_1|C)] \times \prod_{j=2}^{J} [m_1 \cdot P(w_j|D_i) + m_2 \cdot P(w_j|C) + m_3 \cdot P(w_j|w_{j-1}, D_i) + m_4 \cdot P(w_j|w_{j-1}, C)], \quad (1)$$

which can be viewed as a combination of information from a local source (i.e., the document) and a global source (i.e., the large text collection). The unigram and bigram models of the documents and the collection are usually estimated using the maximum likelihood estimation (MLE). The weighting parameters, $m_1,..,m_4$, can be optimized by the expectation-maximization (EM) or minimum classification error (MCE) training algorithms, given a training set of query examples with corresponding query-document relevance information [9].

When the HMM is applied to extractive spoken document summarization, each sentence $S_{i,g}$ of a spoken document D_i is treated as a probabilistic generative model (or HMM) consisting of n-gram distributions for predicting the document, and the terms (or words) in the document D_i are taken as an input observation sequence. In this paper, we only investigate unigram modeling for HMM; thus, the HMM model for a sentence can be expressed as:

$$P_{HMM}(D_i \mid S_{i,g}) = \prod_{w_j \in D_i} [\lambda \cdot P(w_j \mid S_{i,g}) + (1-\lambda) P(w_j \mid C)]^{c(w_j, D_i)}, \quad (2)$$

where λ is a weighting parameter and $c(w_j, D_i)$ is the occurrence count of a term w_j in D_i. In the HMM, the sentence model $P(w_j \mid S_{i,g})$ and the collection model $P(w_j \mid C)$ for each sentence are simply estimated from the sentence itself and a large external text collection, respectively. The weighting parameter λ can be further optimized by taking the document D_i as the training observation sequence and using the following EM training formula:

$$\hat{\lambda} = \frac{\sum_{w_j \in D_i} c(w_j, D_i) \cdot \frac{\lambda \cdot P(w_j \mid S_{i,g})}{\lambda \cdot P(w_j \mid S_{i,g}) + (1-\lambda) \cdot P(w_j \mid C)}}{\sum_{w_l \in D_i} c(w_l, D_i)}. \quad (3)$$

Once the HMM models for the sentences have been estimated, they are used to predict the occurrence probability of the terms in the spoken document. The sentences with the highest probabilities are then selected and sequenced to form the final summary according to different summarization ratios.

2.2 Relevance Model (RM)

In the sentence HMM, as shown in Eq. (2), the sentence model $P(w_j \mid S_{i,g})$ is linearly interpolated with the collection model $P(w_j \mid C)$ to have some probability of generating every term in the vocabulary. However, the true sentence model $P(w_j \mid S_{i,g})$ might not be accurately estimated by MLE, since the sentence only consists of a few terms, and the portions of the terms in the sentence are not the same as the probabilities of those terms in the true model. Therefore, we explore the use of the relevance model (RM) [10, 11], which was originally formulated for IR, to derive a more accurate estimation of the sentence model. In the extractive spoken document summarization task, each sentence $S_{i,g}$ of the document D_i to be summarized has its own associated relevant class $R_{S_{i,g}}$, which is defined as the subset of documents in the collection that are relevant to the sentence $S_{i,g}$. The relevance model of the sentence $S_{i,g}$ is defined as the probability distribution $P(w_j \mid RM_{i,g})$, which gives the probability that we would observe a term w_j if we were to randomly select some document from the relevant class $R_{S_{i,g}}$ and then pick a random term from that document. Once the relevance model of the sentence $S_{i,g}$ has been constructed, it can be used to replace the original sentence model, or it can be combined with the original sentence model to produce a better estimated model. Because there is no prior knowledge about the subset of relevant documents for each sentence $S_{i,g}$, a local feedback-like procedure can be employed by taking $S_{i,g}$ as a query and posing it to the IR system to obtain a ranked list of documents. The top K documents returned by the IR system are assumed to be relevant to $S_{i,g}$, and the relevance model $P(w_j \mid RM_{i,g})$ of $S_{i,g}$ can therefore be constructed by the following equation:

$$P(w_j \mid RM_{i,g}) = \sum_{D_l \in \{\mathbf{D}\}_{\text{Top } K}} P(D_l \mid S_{i,g}) P(w_j \mid D_l), \tag{4}$$

where $\{\mathbf{D}\}_{\text{Top } K}$ is the set of top K retrieved documents; and the probability $P(D_l \mid S_{i,g})$ can be approximated by the following equation using the Bayes' rule:

$$P(D_l \mid S_{i,g}) \approx \frac{P(D_l) P(S_{i,g} \mid D_l)}{\sum_{D_u \in \{\mathbf{D}\}_{\text{Top } K}} P(D_u) P(S_{i,g} \mid D_u)}. \tag{5}$$

A uniform prior probability $P(D_l)$ can be further assumed for the top K retrieved documents, and the sentence likelihood $P(S_{i,g} \mid D_l)$ can be calculated using an equation similar to Eq. (1) if the IR system is implemented with the HMM retrieval model. Consequently, the relevance model $P(w_j \mid RM_{i,g})$ is combined linearly with the original sentence model $P(w_j \mid S_{i,g})$ to form a more accurate sentence model:

$$\hat{P}\left(w_j \mid S_{i,g}\right) = \alpha \cdot P\left(w_j \mid S_{i,g}\right) + (1-\alpha) \cdot P\left(w_j \mid RM_{i,g}\right), \tag{6}$$

where α is a weighting parameter. The final sentence HMM is thus expressed as:

$$\hat{P}_{HMM}\left(D_i \mid S_{i,g}\right) = \prod_{w_j \in D_i} \left[\lambda \cdot \hat{P}\left(w_j \mid S_{i,g}\right) + (1-\lambda) P\left(w_j \mid C\right)\right]^{c\left(w_j, D_i\right)}. \tag{7}$$

Fig. 2 shows a diagram of spoken document summarization using the HMM and RM models.

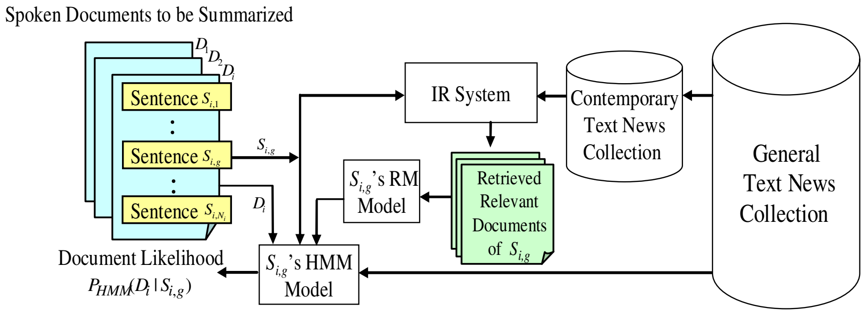

Spoken Documents to be Summarized
D_1
D_2
D_i
Sentence $S_{i,1}$
:
Sentence $S_{i,g}$
:
Sentence S_{i,N_i}
$S_{i,g}$
D_i
IR System
Contemporary Text News Collection
General Text News Collection
$S_{i,g}$'s RM Model
Retrieved Relevant Documents of $S_{i,g}$
$S_{i,g}$'s HMM Model
Document Likelihood
$P_{HMM}(D_i \mid S_{i,g})$

Fig. 2. A diagram of spoken document summarization using the HMM and RM models

3 Experiment Setup

3.1 Speech and Text Corpora

The speech data set was comprised of approximately 176 hours of radio and TV broadcast news documents collected from several radio and TV stations in Taipei between 1998 and 2004 [12]. From them, a set of 200 documents (1.6 hours) collected in August 2001, was reserved for the summarization experiments [1]. The remainder of the speech data was used to train an acoustic model for speech recognition, of which about 4.0 hours of data with corresponding orthographic transcripts was used to bootstrap the acoustic model training, while 104.3 hours of the remaining untranscribed speech data was reserved for unsupervised acoustic model training [13]. The acoustic models were further optimized by the minimum phone error (MPE) training algorithm. A large number of text news documents collected from the Central News Agency (CNA) between 1991 and 2002 (the Chinese Gigaword Corpus released by LDC) was also used. The text news documents collected in 2000 and 2001 were used to train *n*-gram language models for speech recognition; and a subset of about 14,000 text news documents collected in the same period as that of the broadcast news documents to be summarized (August 2001) was used to construct the HMM and RM models.

3.2 Broadcast News Transcription

Front-end processing was performed with the HLDA-based (Heteroscedastic Linear Discriminant Analysis) data-driven Mel-frequency feature extraction approach and further processed by MLLT (Maximum Likelihood Linear Transformation) transformation for feature de-correlation. The speech recognizer was implemented with a left-to-right frame-synchronous Viterbi tree search as well as a lexical prefix tree organization of the lexicon. The recognition hypotheses were organized into a word graph for further language model rescoring. We used a word bigram language model in the tree search procedure and a trigram language model in the word graph rescoring procedure. The Chinese character error rate (CER) for the 200 broadcast news documents reserved for summarization was 14.17%.

3.3 Evaluation Metric

Three subjects were asked to summarize the 200 broadcast news documents, which were to be used as references for evaluation, in two ways:1) to rank the importance of the sentences in the reference transcript of the broadcast news document from the top to the middle; and 2) to write an abstract of the document with a length roughly equal to 25% of the original broadcast news document. Several summarization ratios of the summary length to the total document length [1] were tested. In addition, the ROUGE measure [14, 15] was used to evaluate the performance levels of the proposed models and the conventional models. The measure evaluates the quality of the summarization by counting the number of overlapping units, such as n-grams and word sequences, between the automatic summary and a set of reference (or manual) summaries. ROUGE-N is an n-gram recall measure defined as follows:

$$ROUGE-N=\frac{\sum\limits_{S\in\mathbf{S}_R}\sum\limits_{gram_n\in S}Count_{match}(gram_n)}{\sum\limits_{S\in\mathbf{S}_R}\sum\limits_{gram_n\in S}Count(gram_n)}, \tag{8}$$

where N denotes the length of the n-gram; S is an individual reference (or manual) summary; $\mathbf{S}_R$ is a set of reference summaries; $Count_{match}(gram_n)$ is the maximum number of n-grams co-occurring in the automatic summary and the reference summary; and $Count(gram_n)$ is the number of n-grams in the reference summary. In this paper, we adapted the ROUGE-2 measure, which uses word bigrams as matching units.

4 Experiment Results

4.1 Comparison of HMM and Other Summarization Models

The summarization results obtained by the HMM summarization model using word indexing terms (HMM-1) are shown in the second column of Table 1; and the

Table 1. The results achieved by the HMM and other summarization models under different summarization ratios

Summarization Ratio	HMM-1	HMM-2	VSM	LSA-1	LSA-2	SenSig	Random
10%	0.2989	0.2945	0.2845	0.2755	0.2498	0.2760	0.1122
20%	0.3295	0.3052	0.3110	0.2911	0.2917	0.3190	0.1263
30%	0.3670	0.3334	0.3435	0.3081	0.3378	0.3491	0.1834
50%	0.4743	0.4755	0.4565	0.4070	0.4666	0.4804	0.3096

corresponding ROUGE-2 recall rates are approximately 0.30, 0.33, 0.37, and 0.47 for the summarization ratios 10%, 20%, 30%, and 50%, respectively. The summarization results of the HMM summarization model using syllable indexing terms (HMM-2) are shown in the third column of the table; and it is obvious that the results are comparable to that of the HMM summarization model using word indexing terms. In the following experiments, unless specified otherwise, the HMM model corresponds to the HMM summarization model using word indexing terms. In addition, all the other summarization models discussed in this subsection also use word indexing terms.

We compared the HMM model with the conventional VSM [4] and LSA models. Two variants of LSA, namely, the model mentioned in Section 1 [4] (LSA-1) and the model in [6] (LSA-2), were evaluated. For a spoken document, LSA-2 simply evaluates the score of each sentence based on the norm of its vector representation in the lower L-dimensional latent semantic space. A fixed number of sentences with relatively large scores are therefore selected to form the summary. In the experiments, we set the value of L at 5, the same as that in [6]. The two LSA models were implemented with the MIT SVD Toolkit [16]. We also tried to select indicative sentences from the spoken document based on the sentence significance score (denoted as the SenSig model) [5]. For example, given a sentence $S_{i,g} = \{w_1, w_2, ..., w_r, ..., w_{N_{i,g}}\}$ of length $N_{i,g}$, the sentence significance score is expressed by the following formula:

$$Sig(S_{i,g}) = \sum_{r=1}^{N_{i,g}} [\beta_1 \cdot I(w_r) + \beta_2 \cdot L(w_r)], \tag{9}$$

where $I(w_r)$ is the product of the term frequency (TF) and the inverse document frequency (IDF) of term w_r [17]; $L(w_r)$ is the logarithm of the bigram probability of w_r given its predecessor term w_{r-1} in $S_{i,g}$, which is estimated from a large contemporary text collection; and β_1 and β_2 are tunable weighting parameters. The results for the above models are shown in columns 4 to 7 of Table 1; the results obtained by random selection (Random) are also listed for comparison. We observe that HMM outperforms the VSM, LSA, and SenSig models, which demonstrates that the HMM-based probabilistic ranking model is indeed a good candidate for the extractive spoken document summarization task addressed by this study.

Table 2. The results of combining the HMM and RM models under different summarization ratios; RM was constructed with the IR system using word indexing terms

Summarization Ratio	M_{doc}=5	M_{doc}=10	M_{doc}=15	M_{doc}=20
10%	0.3074	0.3078	0.3078	0.3078
20%	0.3265	0.3284	0.3260	0.3260
30%	0.3667	0.3650	0.3661	0.3676
50%	0.4759	0.4764	0.4762	0.4768

Table 3. The results of combining the HMM and RM models under different summarization ratios; RM was constructed with the IR system using syllable indexing terms

Summarization Ratio	M_{doc}=5	M_{doc}=10	M_{doc}=15	M_{doc}=20
10%	0.3057	0.3111	0.3152	0.3152
20%	0.3254	0.3344	0.3341	0.3332
30%	0.3673	0.3659	0.3659	0.3659
50%	0.4782	0.4770	0.4768	0.4759

4.2 Combination of HMM and RM

As mentioned in Section 2.2, when the HMM is used for summarization, the sentence model $P(w_j \mid S_{i,g})$ might not be accurately estimated, since each sentence of a spoken document consists of only a few words and the portions of words present in the sentence are not necessarily the same as the probabilities of those words in the true model. Therefore, we combine the RM model $P(w_j \mid RM_{i,g})$ with the sentence model $P(w_j \mid S_{i,g})$ to produce a better estimated sentence model, as expressed in Eq. (6). To construct the RM model, each sentence of the spoken document to be summarized is taken as a query and posed to the IR system to obtain a set of M relevant documents from the contemporary text news collection. We implement the IR system with the HMM retrieval model using either words or syllables as the indexing terms. The results of combining the HMM and RM models are shown in Tables 2 and 3. In Table 2, the IR system uses words as the indexing terms to construct the RM model, while, in Table 3, syllables are adopted as the indexing terms for the IR system. Each column in the tables indicates the number of relevant documents (M_{doc}) returned by the IR system for construction of the RM model.

A number of conclusions can be drawn from the results. First, the combination of HMM and RM boosts the summarization performance when the summarization ratios are low (e.g., 10%), while the gains are almost negligible at higher summarization ratios. Second, the RM model constructed based on the IR system using syllables as indexing terms is better than that based on the IR system using words as indexing terms. One possible reason is that the automatic transcript of a sentence in a broadcast news document often contains speech recognition errors and, in Chinese, syllable

Table 4. The results of combining the HMM and RM models, using syllable indexing terms; the RM model was constructed with the IR system using syllable indexing terms

Summarization Ratio	M_{doc}=5	M_{doc}=10	M_{doc}=15	M_{doc}=20
10%	0.3190	0.3276	0.3285	0.3285
20%	0.3327	0.3414	0.3439	0.3439
30%	0.3473	0.3544	0.3542	0.3542
50%	0.4735	0.4750	0.4724	0.4724

Table 5. The results of combining the HMM and RM models, using both word and syllable indexing terms; the RM model was constructed with the IR system using syllable indexing terms

Summarization Ratio	M_{doc}=5	M_{doc}=10	M_{doc}=15	M_{doc}=20
10%	0.3305	0.3285	0.3335	0.3352
20%	0.3411	0.3391	0.3442	0.3468
30%	0.3641	0.3641	0.3612	0.3645
50%	0.4809	0.4816	0.4781	0.4782

accuracy is always higher than word accuracy. Therefore, the IR system that uses syllables as indexing terms might retrieve a set of more relevant documents than the system using single words. Finally, the summarization performance seems to become saturated when the IR system returns 15 relevant documents for construction of the RM model.

4.3 Information Fusion Using Word- and Syllable-Level Indexing Terms

The summarization results reported in Sections 4.1 and 4.2 were obtained in such a way that the summarization models were only implemented with words as the indexing terms, although the IR system used to construct the RM model can use either words or syllables as indexing terms. Hence, we also implemented the models by using syllables as indexing terms. The summarization results of combining the HMM and RM models and using syllable indexing terms, are shown in Table 4. In this case, the RM model was also constructed with the IR system using syllable indexing terms. Compared with the results in Table 3, the summarization model implemented with syllable indexing terms is considerably better than the one implemented with word indexing terms, especially at lower summarization ratios. Finally, the results derived by combining the HMM and RM models, as well as by using both word and syllable indexing terms, are shown in Table 5. Compared with the results in Table 4, the fusion of these two kinds of indexing information clearly yields additional performance gains. This is because word-level indexing terms contain more semantic information, while syllable-level indexing terms are more robust against errors in speech recognition. Thus, combining these two kinds of indexing terms for the Chinese spoken document summarization task is effective.

5 Conclusions

We have presented an HMM-based probabilistic model for extractive Chinese spoken document summarization. The model's capabilities were verified by comparison with other summarization models. Moreover, the RM model of each sentence of a spoken document to be summarized was integrated with the sentence HMM model for better model estimation. The experiment results are very promising. In our current implementation, the relevant model trained on relevant documents retrieved for a sentence from a contemporary text collection is integrated with the sentence HMM. These relevant documents can be used to train the sentence HMM directly. We believe this is a more effective way to utilize relevant documents.

Acknowledgments. This work was supported in part by the National Science Council, Taiwan, under Grants: NSC95-2221-E-003-014-MY3 and NSC95-2422-H-001-008.

References

1. L.S. Lee and B. Chen, "Spoken Document Understanding and Organization," *IEEE Signal Processing Magazine*, Vol. 22, No. 5 (2005) 42-60
2. S. Furui, T. Kikuchi, Y. Shinnaka and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," *IEEE Trans. on Speech and Audio Processing*, Vol. 12, No. 4 (2004) 401-408
3. I. Mani and M. T. Maybury, Eds. Advances in Automatic Text Summarization. Cambridge. MA: MIT Press (1999)
4. Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *Proc. ACM SIGIR Conference on R&D in Information Retrieval*, (2001) 19-25
5. J. Goldstein et al., "Summarizing text documents: sentence selection and evaluation metrics," in *Proc. ACM SIGIR Conference on R&D in Information Retrieval* (1999) 121-128
6. M. Hirohata et al., "Sentence Extraction-based Presentation Summarization Techniques and Evaluation Metrics," in Proc. IEEE International Conference on Acoustics, Speech, and Signal processing (2005) 1065-1068
7. K. Koumpis and S. Renals, "Automatic Summarization of Voicemail Messages Using Lexical and Prosodic Features," *ACM Transaction. on Speech and Language Processing*, Vol. 2, No.1 (2005) 1-24
8. S. Maskey and J. Hirschberg, "Comparing Lexical, Acoustic/Prosodic, Structural and Discourse Features for Speech Summarization," in *Proc. European Conference on Speech Communication and Technology* (2005) 621-624
9. B. Chen, H.M. Wang, L.S. Lee, "A Discriminative HMM/N-Gram-Based Retrieval Approach for Mandarin Spoken Documents," *ACM Transactions on Asian Language Information Processing*, Vol. 3, No. 2 (2004) 128-145
10. Croft, W.B., Lafferty, J. (Eds.). Language Modeling for Information Retrieval. Kluwer-Academic Publishers (2003)

11. M. D. Smucker et al., “Dirichlet Mixtures for Query Estimation in Information Retrieval,” CIIR Technical Report, Center for Intelligent Information Retrieval, University of Massachusetts (2005)
12. B. Chen et al., “Speech Retrieval of Mandarin Broadcast News via Mobile Devices,” in *Proc. European Conference on Speech Communication and Technology* (2005) 109-112
13. B. Chen et al., “Lightly Supervised and Data-Driven Approaches to Mandarin Broadcast News Transcription,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal processing* (2004) 777-780
14. C.Y. Lin, “ROUGE: Recall-Oriented Understudy for Gisting Evaluation,” (2003) Available from: <http://www.isi.edu/~cyl/ROUGE/>
15. C.-Y. Lin, “Looking for a few good metrics: ROUGE and its evaluation,” Working Notes of NTCIR-4 (Vol. Supl. 2) (2004) 1-8
16. D. Rohde. Doug Rohde's SVD C Library, Version 1.34 (2005) Available from: <http://tedlab.mit.edu:16080/~dr/SVDLIBC/>
17. R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. Addison-Wesley (1999)

Meeting Segmentation Using Two-Layer Cascaded Subband Filters

Manuel Giuliani, Tin Lay Nwe, and Haizhou Li

Institute for Infocomm Research
Republic of Singapore
manuel@manuelgiuliani.de, tlnma@i2r.a-star.edu.sg, hli@i2r.a-star.edu.sg

Abstract. The extraction of information from recorded meetings is a very important yet challenging task. The problem lies in the inability of speech recognition systems to be directly applied onto meeting speech data, mainly because meeting participants speak concurrently and head-mounted microphones record more than just their wearers' utterances - crosstalk from his neighbours are inevitably recorded as well. As a result, a degree of preprocessing of these recordings is needed. The current work presents an approach to segment meetings into four audio classes: *Single speaker*, *crosstalk*, *single speaker plus crosstalk* and *silence*. For this purpose, we propose Two-Layer Cascaded Subband Filters, which spread according to the pitch and formant frequency scales. This filters are able to detect the presence or absence of pitch and formants in an audio signal. In addition, the filters can determine how many numbers of pitches and formants are present in an audio signal based on the output subband energies. Experiments conducted on the ICSI meeting corpus, show that although an overall recognition rate of up to 57% was achieved, rates for crosstalk and silence classes are as high as 80%. This indicates the positive effect and potential of this subband feature in meeting segmentation tasks.

1 Introduction

Meetings are an important part of everyday work life. Many spend more time in meetings, where important goals and new strategies are discussed and determined, than on their desks. It is therefore desirable to extract the contents of a meeting and conserve them for future work or for purposes of proof. Automatic speech recognition (ASR), for example, seems to be a good tool to extract at least the textual content of a meeting. Unfortunately the recognition of speech in recorded meetings is a difficult task. Meeting participants speak naturally (i.e. use natural language), interrupt each other, talk at the same time and also use ungrammatical or incomplete sentences. In turn, these meeting conditions and norms, negatively affect ASR recognition rate. That is why some form of preprocessing of the recorded meetings is needed, among other things, to determine how many persons had been speaking at any one time in the meeting.

There have been some attempts at preprocessing of meetings. Dielmann and Renals [1] tried to segment meetings automatically into a set of social actions

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 672–682, 2006.

such as *monologue*, *discussion* and *presentation.* For that purpose, they combined prosodical, lexical and speaker activity features to train and test a dynamic Bayesian Network model. With the so-called speaker activity feature, one can estimate which direction of the meeting room the speech recorded at a time is coming from. Therefore a microphone array was used to simulate a steerable directional microphone. Their experiments achieved a recognition rate of 92.9% and were conducted on the M4 corpus, which had been recorded at the IDIAP Research Institute. The M4 corpus contains 53 short meetings, recorded using lapel microphones for each meeting participant, and an eight element circular microphone array. However the lexical features that were used were based on human-generated word-level transcription of the meetings, entailing the employment of significant manual effort and that the segmentation process cannot be done automatically.

Wrigley et al. [2] segmented meetings into four different audio classes, namely *single speaker* (S), *crosstalk* (C), *speaker plus crosstalk* (SC) and *silence* (SIL). Crosstalk occurs when the lapel microphones or head-mounted microphones of meeting participants record not only their wearers' utterances, but also spoken comments from their neighbours. To discriminate the four different classes, Wrigley et al. analyzed several features on their efficiency for the task. Besides the classical speech processing features like MFCCs, Energy and Zero Crossing Rate, they also tested other features which had been proven to work well in similar tasks. These features include the following: Kurtosis, Fundamentalness, Spectral Autocorrelation Peak-Valley Ratio, Pitch Prediction, features derived from genetic programming and cross-channel correlation. After feature evaluation, they presented a system which consisted of a multistream ergodic Hidden Markov Model (eHMM) and a rule-based post processor to test the feature sets they had first found. They reported high average recognition rates of 76.5% for the speaker alone class, and 94.1% for the crosstalk alone class, but very low recognition rates for single speaker plus crosstalk and silence.

In the current work, we implement Two-Layer Cascaded Subband Filters (TLCSF) for meeting segmentation. The filters are able to extract the information of number of speakers based on the pitch and formant information. We combine this feature with other features which had been reported in [2] to classify the audio classes S, C, SC and SIL with higher accuracy. We trained Gaussian Mixture Models (GMM) for the four classes and linked them to an ergodic HMM. Experiment results show that our recognition rates are significantly higher for the classes SC and SIL.

The remaining of this paper is organized as follows: Section 2 describes the International Computer Science Institute (ICSI) meeting corpus which was used in our experiments. Following that is a presentation of our ergodic Hidden Markov Model in section 3, and an explanation of the acoustic parameters used in section 4. With the model and features in place, a range of experiments were conducted and are presented and discussed in section 5. Finally section 6 contains a conclusion of the current work as well as an outline of a few possibilities for future improvements.

2 Corpus

The ICSI Meeting Corpus consists of 75 meetings, which were recorded during the years 2000 - 2002 at the International Computer Science Institute (ICSI) in Berkeley, California. The meetings were not restricted by any guidelines, that means the recording sessions were held during normal meetings, which would have been conducted regardless of the recordings. In these recording sessions, every meeting participant wore either a head-mounted or a lapel microphone. At the same time, the meeting was recorded by six table microphones of different qualities. The meeting lengths range between 17 and 103 minutes and the corpus contains 72 hours of recorded speech in total. The data were collected at a 48 kHZ sample-rate, which was downsampled to 16 kHz. The audio files for each meeting are provided as separate time-synchronous recordings for each channel, encoded as 16-bit linear wave files and saved in the NIST sphere format. For each meeting, a time-tagged word-level transcript is available, which also contains meta information about its meeting participants and the hardware used for the session. A full description of the corpus can be found in [3]. From the corpus we chose 30 meetings, of which the data from 11 meetings were used to train the ergodic Hidden Markov Model and the data from the remaining 19 meetings were used for testing purposes.

3 Model

The model used in this work is an ergodic Hidden Markov Model (eHMM), which is made up of four GMMs, one for each of the four classes - S, C, SC and SIL - that we want to detect. The term *ergodic* refers to the fact that all four states of the HMM are linked together, such that every state is reachable from any other state and by itself, as illustrated in Figure 1. A GMM is defined by

$$\sum_{i=1}^{G} p_i \Phi_i(X, \mu_i, \Sigma_i) \tag{1}$$

where X is the feature vector and G is the number of Gaussian densities Φ_i. Every Φ_i has a mean vector μ_i, a covariance matrix Σ_i and a mixing coefficient p_i.

Every GMM was trained with the expectation-maximization algorithm, as it is implemented in the Hidden Markov Toolkit (HTK). The training data was extracted from 11 meetings [1] of the corpus. For each of the four classes, one million feature vectors were chosen randomly from the data. The number of mixtures per GMM varied and were chosen according to the values mentioned in [2]. The number of mixtures for the classes S and SC was set to 20, and to 5 and 4 for the classes C and SIL respectively. After training, the four GMMs were linked with transitions, such that every GMM is reachable from any other

[1] Training data was taken from the following meetings: Bed006, Bed008, Bed010, Bed011, Bmr001, Bmr005, Bmr006, Bmr014, Bmr024, Bro007, and Bro012.

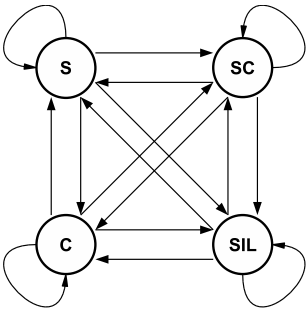


Fig. 1. Ergodic Hidden Markov Model, comprising four GMMs

GMM. The arcs between the GMMs were provided with transition probabilities, which were computed from the meeting transcripts.

4 Acoustic Parameters

The expressiveness of the acoustic parameters has direct impact on segmenting audio into different classes. In addition to short-time spectral information, we integrate pitch and formant information into our acoustic features. We propose Two-Layer Cascaded Subband Filters (TLCSF), which spread according to the pitch and formant frequency ranges. This filters are able to detect the presence or absence of pitch and formants in an audio segment. Furthermore, the filters can determine how many number of pitches or formants are present in an audio segment from the output subband energies. We transform these subband energies into cepstral coefficients for statistical modeling. The cepstral coefficients are used, because they have been proven to be robust in audio and speech recognition [7].

4.1 Acoustic Characteristics and Audio Classes

Before computing the features, we examine the significant characteristics possessed by each audio class. The signal strengths of the classes S (speaker alone) and SC (speaker plus crosstalk) are higher than those of class C (crosstalk alone) and class SIL (silence). In addition, the numbers of pitches and formants present in class S and class SC are different. The audio segment of class S has only one pitch or formant. However, the audio segment of class SC can present more than one pitch and formant. Furthermore, pitch and formant are not present in class SIL. Therefore, the acoustic features to identify these 4 audio classes (S, SC, C and SIL) should reflect the information on 1) signal strength, 2) the presence or absence of pitches and formants and 3) how many numbers of pitches and formants are present in an audio segment. To this end, we propose Two-Layer Cascaded Subband Filters to capture the above information from an audio signal.

4.2 Two-Layer Cascaded Subband Filters (TLCSF)

We propose Two-Layer Cascaded Subband Filters, shown in Figure 2, to capture the information of pitch, formant and signal strength. The filter has two cascaded layers. The first layer has overlapped rectangular filters. For each filter in the first layer, there are 5 non-overlapped rectangular filters of equal bandwidth in the second layer. The first filter of the first layer has a bandwidth spanned between 65Hz and 250Hz. This bandwidth covers the pitch of male and female in general [8]. This filter is able to determine the information on 1) presence or absence of pitch, and 2) number of pitches in an audio segment. Details on how the filter captures pitch information will be discussed in later paragraphs. Bandwidths of the following filters cover F1 (First formant), F2 (Second formant) and F3 (Third formant) of 15 selected English vowels [5]. Each of these filters determines the information on 1) presence or absence of formant, and 2) number of formants in an audio segment. To this end, we need to implement 1 filter for pitch and 45 filters for F1, F2 and F3 of each of the 15 vowels. Note that the formants of some vowels (example, First formants of vowels [aÈ] and [aØ]) overlap each other. Hence, we need to implement only one filter for these overlapped formants. Finally, we have 1 filter for the pitch (F0) and 40 filters for the formants (F1, F2 and F3). In Total, we implement 41 filters in the first layer. The center frequencies and bandwidths of all filters are listed in Table 1.

We have 41 filters in the first layer. For each filter of the first layer, we have 5 non-overlapped filters in the second layer. Hence, we have a total of 205 (41 x 5) filters in the second layer. The range of our subband filters is from 65 Hz to 3.2kHz.

The upper panels of Figure 3 (a), (b), (c) and (d) represent the signals of the four audio classes S, SC, C and SIL in the pitch frequency range (65Hz to 250Hz). As can be seen in the figures, the audio classes S and SC have the strongest signal strength of all four classes. In addition, only one pitch is present in audio class S and two pitches are present in the class SC, while no pitch is present in the class SIL. The Two-Layer Cascaded Subband Filter captures this information as follows.

The pitch information is captured by TLCSF for the four audio classes which are presented in Figures 3 (a), (b) (c) and (d). In each figure, the signal in the upper panel is passed through the TLCSF filters shown in the middle panel. Then, the output amplitudes of the five subband filters are computed and shown in the lower panel. As can be seen in the figures, the number of local maxima in the lower panel is the number of pitches present in the audio signal. Since TLCSF

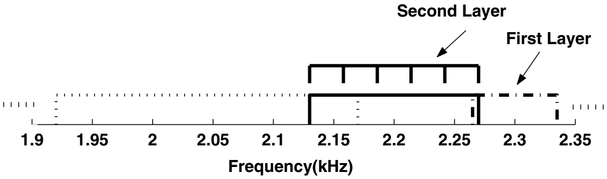


Fig. 2. A bank of Two-Layer Cascaded Subband Filters

Table 1. Center Frequencies (CF) and Bandwidths (BW) of the 41 subbands in the first layer

No	Type	Vowel	CF(Hz)	BW (Hz)
1	F0	-	157.5	185
2	F1	[i]	300	72.5
3	F1	[u]	335	95
4	F1	[eÈ]	405	212.5
5	F1	[È]	435	120
6	F1	[ÿ]	445	150
7	F1	[oÈ]	455	260
8	F1	[Ø]	475	130
9	F1	[oØ]	495	170
10	F1	[aÈ], [a]	530	345
11	F1	[E]	575	150
12	F1	[O]	615	120
13	F1	[U]	620	80
14	F1	[Ï]	635	100
15	F1	[A]	700	130
16	F2	[oØ]	1000	270
17	F2	[O]	1015	150
18	F2	[u]	1075	460
19	F2	[E]	1085	360
20	F2	[Ø]	1140	180
21	F2	[A], [U]	1220	70
22	F2	[ÿ]	1290	100
23	F2	[oÈ]	1390	910
24	F2	[aÈ], [ÿ]	1540	765
25	F2	[Ï]	1575	295
26	F2	[E]	1605	240
27	F2	[È]	1700	300
28	F2	[eÈ]	1870	400
29	F2	[i]	2045	250
30	F3	[u]	2200	140
31	F3	[oØ]	2300	70
32	F3	[Ø]	2370	120
33	F3	[oÈ]	2425	195
34	F3	[Ï], [aØ]	2450	360
35	F3	[E]	2515	230
36	F3	[aÈ]	2525	250
37	F3	[U]	2550	140
38	F3	[eÈ]	2560	280
39	F3	[È],[O]	2585	170
40	F3	[A]	2600	160
41	F3	[i]	2960	400

includes subbands for pitch and formant frequency ranges, these subbands work together to capture the pitch and formant information of the signal.

As mentioned above, formants of some vowels overlap each other. In Table 1, the filters with numbers 10, 21, 24, 34 and 39 are for 2 overlapped formants. Each of these filters covers the formants of two vowels. These filters can wrongly classify S as SC. The reason can be explained as follows: Let us assume, a speech segment of class S includes two vowels, [aÈ] and [aØ], which formants overlap in filter number 10. Two local maxima (two formants) can be present in the output of filter number 10 similar to the one shown in the lower panel of Figure 3(b). As we discussed above, if an audio segment has two local maxima, the classifier classifies the segment as 'SC'. Hence, we need to make sure that only one local maximum presents a segment of class 'S'. To this end, we choose an analysis frame length that covers the duration of only one vowel. For this reason, we choose the frame length of 60ms which is less than the average vowel duration [4].

4.3 Computation of TLCSF Coefficients

The speech signal was divided into frames of 60ms with 10ms overlapping. Each frame was multiplied by a Hamming window to minimize signal discontinuities at the end of each frame. Next, fast fourier transform (FFT) was applied, and

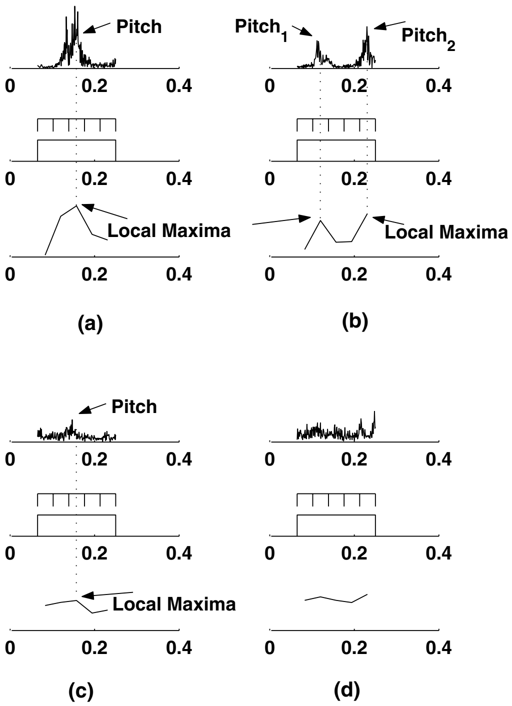


Fig. 3. Capturing pitch information and TLCSF subband filtering: (a) Speaker alone - only one pitch and strong signal strength. (b) Speaker plus crosstalk - two pitches and strong signal strength. (c) Crosstalk alone - only one pitch and weak signal strength. (d) Silence - no pitch and weak signal strength. In each figure, the upper panel shows the signal. The middle panel presents the frequency response of the TLCSF subband filters. The lower panel demonstrates the output of the TLCSF subband filters. The filters capture the information on 1) presence or absence of pitch, and 2) number of pitches in the signal by detecting the local maxima.

following that, the audio frame was passed through a bank of cascaded subband filters and the log energy of each of 205 bands in the second layer was computed. Finally, a total of 40 Pitch and Formant Frequency Cepstral Coefficients (PF-FCC) was computed from log energies using Discrete Cosine Transform [9] for each audio frame.

In Figure 4, example frames for the four classes S, SC, C and SIL and their corresponding feature vectors are illustrated. It can be seen clearly that the values of the feature vectors can be used to discriminate between the classes. For each class in Figure 4, two panels are shown. The top panels illustrate the signal in time domain. Their corresponding subband feature vectors with 205 values are shown in the lower panel. Please note that the scales for the signals of S/SC and C/SIL differ for illustration purposes.

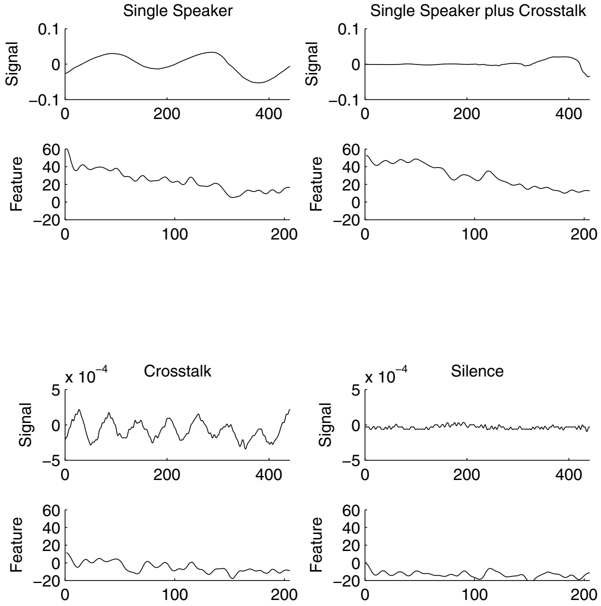


Fig. 4. Illustration of signals and feature vectors for all four audio classes

4.4 Features from Other Work

In addition to the PFFCC we introduced, seven other features were added. These have been shown to give good results in the meeting segmentation processes in [2]. Each of the following features was computed over a 16ms Hamming window with a frame-shift of 10ms.

- *Cross-channel Correlation* (CCC). The CCC is the maximum of the cross-channel correlation between a particular channel and any other channel. It was computed at any time of the signal. For each set of correlation values for any channel, the mean CCC, maximum normalized CCC and mean and maximum spherically normalized CCC was computed.
- *Kurtosis* (KU). Kurtosis is the fourth-order moment of a signal divided by the square of its second-order moment.
- *Log Energy* (LE).

5 Experiments

For the experiments, data from 19 of the ICSI meetings[2] were used. We tried combinations of different features to study the effects of the following parameters: Window length of the PFFCC feature, reduction of PFFCC features by dct, as

[2] The data of the following meetings were used: Bed015, Bed017, Bmr002, Bmr007, Bmr008, Bmr009, Bmr013, Bmr018, Bmr022, Bmr026, Bmr027, Bro008, Bro013, Bro014, Bro015, Bro017, Bro018, Bro023, Bro026.

well as a combination of the PFFCC features with the parameters *Cross-channel Correlation*, *Kurtosis* and *Log Energy*. As mentioned in Section 4.2, 60ms is a suitable window length for this task since this length is the average duration of a vowel. However, we would like to see the effect on the system performance using a shorter window length (example, 20ms). The reason is that a shorter window length could be a better choice to make sure that the audio segment includes only one vowel. Hence, we use window lengths of 20ms and 60ms to extract features. All these parameters led to six feature sets which are listed in Table 2. According to the window length and the number of features, the sets are named 20-41, 20-46, 20-211, 60-41, 60-46 and 60-211.

Table 2. Composition of feature sets

Window	No.	No. PFFCC		CCC	KU	LE
Length	Total	40	205			
20	41	●				●
20	46	●		●	●	●
20	211		●	●	●	●
60	41	●				●
60	46	●		●	●	●
60	211		●	●	●	●

The feature vectors from all the test meetings were extracted and labeled. Then recognition tests were made using the HTK. As we were interested to study only the effects of the feature combinations, no smoothing strategy was applied to the outgoing streams of recognition results.

In Table 3, we report the overall recognition results for two of the meetings (namely Bed015 and Bmr002) to show the performance of the different feature sets. It can be seen that the 41-dimensional feature sets, which contain the reduced PFFCC features and log energy, clearly outperform the remaining sets. But it should be noted that the low performance of the other feature sets may be due to a problem of normalization, which can be solved in future studies. In addition, we found that a 60ms window length performs better than a 20ms window. The reason for that is that a short window can not show a significant spectral difference between the different audio classes.

Table 3. Overall recognition results for meetings Bed015 and Bmr022

Set	Bed015	Bmr022
20-41	**44.5%**	**46.0%**
20-46	33.9%	39.3%
20-211	22.7%	22.9%
60-41	**47.9%**	**52.7%**
60-46	28.6%	36.3%
60-211	19.0%	17.0%

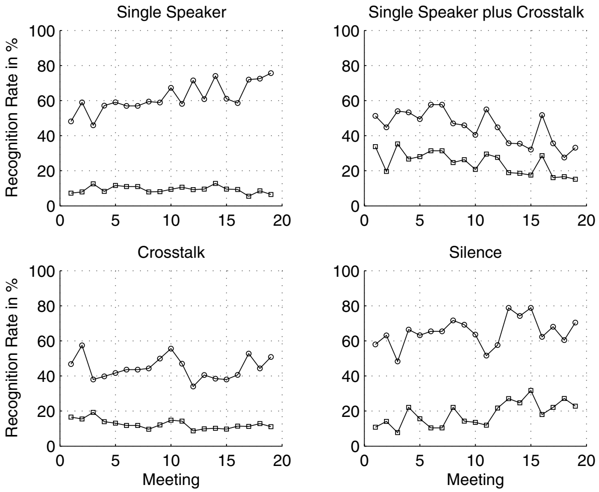


Fig. 5. Recognition rates (upper line with circles) and false positive rates (lower line with squares) for all 19 test meetings using feature set 60-41. Each of the circles and squares stands for the recognition rate, and false positive rate accordingly, of one meeting.

Figure 5 displays the recognition rate (line with circles) and the false positive rate (line with squares) for the single states of all meetings for the 60-41 feature set, which performed best. The recognition rate denotes the percentage of correctly classified frames, while the false positive rate is specified as the proportion of negative instances that were erroneously reported as being positive.

These results also show that the recognition rate for the classes S and SIL are much higher than for the classes C and SC. Since our system aims to be used in the preprocessing of meetings for ASR systems, these results are very useful as they denote that a rather accurate detection of single speaker frames is possible and achievable. This indicates that the PFFCC feature is indeed suitable for the detection of several speakers and deserves further investigation. Our results can't be compared directly to the ones reported in [2], because on the one hand we used a slightly different set of the recorded meetings for training and testing. And on the other hand Wrigley et al. don't report their recognition results before applying a smoothing strategy.

6 Conclusions

In this paper, we presented a system for meeting segmentation, which segments recorded meetings into the four audio classes: *Single speaker*, *crosstalk*, *single speaker plus crosstalk* and *silence*. For that purpose we trained several ergodic Hidden Markov Models with different feature sets, which were made up of a feature that had been computed with two layers of subband-based filters, plus several other features that had been reported in other publications. Experiment results show that the performance of our system is effective for the single speaker and silence classes. Upcoming tasks to improve the recognition rate for the other classes can include normalization of the feature sets and implementing different models.

Acknowledgments

The authors like to thank Tomi Kinnunen for his help with the subband feature reduction.

References

[1] Alfred Dielmann and Steve Renals, "Multistream Dynamic Bayesian Network for Meeting Segmentation", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP2006), May 14-19, 2006, Toulouse, France

[2] Stuart N. Wrigley, Guy J. Brown, Vincent Wan and Steve Renals, "Speech and Crosstalk Detection in Multichannel Audio", IEEE Transactions on Speech and Audio Processing, Vol. 13, No. 1, January 2005

[3] Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke and Chuck Wooters, "The ICSI Meeting Corpus", Proc. ICASSP, 2003, pp. 364 - 367

[4] Xue Wang, Louis C.W. Pools and Louis F.M. ten Bosch, "Analysis of Context-Dependent Segmental Duration for Automatic Speech Recognition", International Conference on Spoken Language Processing (ICSLP), 1996, 1181-1184

[5] Dennis H. Klatt, "Software for a Cascade/Parallel Formant Synthesizer", J. Acoust. Soc. Am. 67, 971-995., 1980

[6] Haizhou Li and Tin Lay Nwe, "Vibrato-Motivated Acoustic Features for Singer Identification", Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, May 14-19, 2006, Toulouse, France.

[7] L. R. Rabiner, B. H. Juang, "Fundamentals of Speech Recognition", Prentice Hall, Englewood Cliffs, N.J, 1993

[8] G. Fant, "Speech Sounds and Features" Cambridge: MIT Press, MA, 1973

[9] C. Becchetti, L. P. Ricotti, "Speech Recognition Theory and C++ Implementation" New York: John Wiley & Sons, 1998

A Multi-layered Summarization System for Multi-media Archives by Understanding and Structuring of Chinese Spoken Documents

Lin-shan Lee, Sheng-yi Kong, Yi-cheng Pan,
Yi-sheng Fu, Yu-tsun Huang, and Chien-Chih Wang

Speech Lab, College of EECS National Taiwan University, Taipei
lslee@gate.sinica.edu.tw

Abstract. The multi-media archives are very difficult to be shown on the screen, and very difficult to retrieve and browse. It is therefore important to develop technologies to summarize the entire archives in the network content to help the user in browsing and retrieval. In a recent paper [1] we proposed a complete set of multi-layered technologies to handle at least some of the above issues: (1) Automatic Generation of Titles and Summaries for each of the spoken documents, such that the spoken documents become much more easier to browse, (2) Global Semantic Structuring of the entire spoken document archive, offering to the user a global picture of the semantic structure of the archive, and (3) Query-based Local Semantic Structuring for the subset of the spoken documents retrieved by the user's query, providing the user the detailed semantic structure of the relevant spoken documents given the query he entered. The Probabilistic Latent Semantic Analysis (PLSA) is found to be helpful. This paper presents an initial prototype system for Chinese archives with the functions mentioned above, in which the broadcast news archive in Mandarin Chinese is taken as the example archive.

1 Introduction

In the future network era, the digital content over the network will include all the information activities for human life. Apparently, the most attractive form of the network content will be in multi-media including speech information, and such speech information usually tells the subjects, topics and concepts of the multi-media content. As a result, the spoken documents associated with the network content will become the key for retrieval and browsing. However, unlike the written documents with well structured paragraphs and titles, the multi-media and spoken documents are both very difficult to retrieve or browse, since they are just audio/video signals, very difficult to be shown on the screen, and the user can not go through each of them from the beginning to the end during browsing. As a result, it will be very important to develop a set of technologies to summarize the entire archives of the spoken documents to help the user to browse and retrieve the multi-media/spoken documents [1,2]. Such summarization technologies for the entire archives at least need a few key elements:

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 683–692, 2006.

information extraction (to extract key information from the spoken documents), document archive structuring (to organize the archive of spoken documents into some form of hierarchical structures) and query-based (able to respond to the user's query to offer the information about a subset of the archive relevant to user's interest).

Note that while some of the technologies mentioned above have been studied or explored to a good extent, most of the work has been performed independently within individual scopes. Great efforts have been made to try to integrate several of these technologies for a specific application , and several well-known research projects have been successfully developed towards this goal. Examples include the Informedia System at Carnegie Mellon University [3], the Multimedia Document Retrieval Project at Cambridge University [4], the Rough'n'Ready System at BBN technologies [5], the Speech Content-based Audio Navigator (SCAN) System at AT&T Labs-Research [6], the Broadcast News Navigator at MITRE Corporation [7], the SpeechBot Audio/Video Search System at Hewlett-Packard (HP) Labs [8], the National Project of Spontaneous Speech Corpus and Processing Technologies of Japan [9], and the NewsBlaster project of Columbia University [10].

In a recent paper [1] we proposed a complete set of multi-layered technologies to handle at least some of the above issues: (1) Automatic Generation of Titles and Summaries for each of the spoken documents, (2) Global Semantic Structuring of the entire spoken document archive, and (3) Query-based Local Semantic Structuring for the subset of the spoken documents retrieved by the user's query. All of the above have to do with the analysis of the semantics carried by the spoken documents. Also we proposed that the Probabilistic Latent Semantic Analysis (PLSA) recently developed for semantic analysis is very helpful [1]. In this paper we present an initial prototype system for the functionalities mentioned above, in which the broadcast news archive in Mandarin Chinese is taken as the example archive.

2 Proposed Approaches

2.1 Probabilistic Latent Semantic Analysis (PLSA)

The set of documents $\{d_i, i = 1, 2, \ldots, N\}$ have been conventionally analyzed by the terms $\{t_j, j = 1, 2, \ldots, L\}$ they may include, usually with statistical approaches. In recent years, efforts have also been made to establish a probabilistic framework for such purposes with improved model training algorithms, of which the Probabilistic Latent Semantic Analysis (PLSA)[11] is often considered as a representative. In PLSA, a set of latent topic variables is defined, $\{T_k, k = 1, 2, \ldots, K\}$, to characterize the "term-document" co-occurrence relationships.Both the document d_i and the term t_j are assumed to be independently conditioned on an associated latent topic T_k. The conditional probability of a document d_i generating a term t_j thus can be parameterized by

$$P(t_j|d_i) = \sum_{k=1}^{K} P(t_j|T_k)P(T_k|d_i). \tag{1}$$

Notice that this probability is not obtained directly from the frequency of the term t_j occurring in d_i, but instead through $P(t_j|T_k)$, the probability of observing t_j in the latent topic T_k, as well as $P(T_k|d_i)$, the likelihood that d_i addresses the latent topic T_k. The PLSA model can be optimized with the EM algorithm by maximizing a carefully defined likelihood function [11].

2.2 Automatic Generation of Titles and Summaries for Spoken Documents

The titles exactly complement the summaries for the user during browsing and retrieval. The user can easily select the desired document with a glance at the list of titles. He can then look through or listen to the summaries in text or speech form for the titles he selected.

Substantial efforts have been made in automatic generation of titles and summaries for spoken documents [12,13,14,15,16]. In this research, it was found that the topic entropy of the terms estimated from probabilities obtained in PLSA analysis is very useful in finding the key terms of the spoken documents in automatic generation of titles and summaries [17][18]. The topic entropy of a term t_j is evaluated from the topic distribution $\{P(T_k|t_j), k = 1, 2, \dots, K\}$ of the term obtained from PLSA and defined as:

$$EN(t_j) = -\sum_{k=1}^{K} P(T_k|t_j) \log P(T_k|t_j) \tag{2}$$

Clearly, a lower entropy implies the term carries more topical information for a few specific latent topics, thus is more significant semantically.

In this research, it was found that the sentences selected based on the topic entropy can be used to construct better summaries for the spoken documents [17]. For title generation, a new delicate scored Viterbi approach was developed in this research based on the concept of the previously proposed statistical translation approach [13]. In this new approach, the key terms in the automatically generated summaries are carefully selected and sequenced by a Viterbi beam search using three sets of scores. This new delicate scored Viterbi approach was further integrated with the previously proposed adaptive K-nearest neighbor approach [19] to offer better results [18].

2.3 Global Semantic Structuring for the Spoken Document Archive

The purpose of global semantic structuring for spoken document archives is to offer an overall knowledge of the semantic content of the entire spoken document archive in some form of hierarchical structures with concise visual presentation to help the user to browse across the spoken documents efficiently. In this research, we developed successfully a new approach to analyze and structure the topics of spoken documents in an archive into a two-dimensional tree structure or a multi-layer map for efficient browsing and retrieval [20]. The basic approach used is based on the PLSA concept. In the constructed two-dimensional tree structure, the spoken documents are clustered by the latent topics they primarily address, and the clusters are organized as a two-dimensional map. The nodes on the map represent the clusters, each labeled by several key terms with the highest scores for the cluster. The nodes are organized on the map in such a way that the distances between nodes have to do with the relationships between the topics of the clusters, i.e., closely located nodes represent clusters with closely related topics. Every node can then be expanded into another two-dimension map in the next layer with nodes representing finer topics. In this way the entire spoken archive can be structured into a two-dimensional tree, or a multi-layered map, representing the global semantics of the archive [20]. This is very useful for browsing and retrieval purposes.

2.4 Query-Based Local Semantic Structuring for Spoken Documents Relevant to User's Interests

The global semantic structure mentioned above is very useful, but not necessarily good enough for the user regarding his special information needs , very often represented by the query he entered to the information retrieval engine. The problem is that the query given by the user is usually very short and thus not specific enough, and as a result a large number of spoken documents are retrieved, including many noisy documents retrieved due to the uncertainty in the spoken document retrieval. However, as mentioned above, the spoken documents are very difficult to be shown on the screen and very difficult to browse. The large number of retrieved spoken document therefore becomes a difficult problem. It is thus very helpful to construct a local semantic structure for the retrieved spoken documents for the user to identify what he really needs to go through or to specify what he really wish to obtain. This semantic structure is localized to user's query, constructed from those retrieved documents only, thus needs to be much more delicate over a very small subset of the entire archive. This is why the global semantic structure proposed above in section 2.3 cannot be used here. Instead in this research we propose to construct a very fine topic hierarchy for the localized retrieved documents. Every node on the hierarchy represents a small cluster of the retrieved documents, and is labeled by a key term as the topic of the cluster. The user can then click on the nodes or topics to select the documents he wishes to browse, or to expand his query by adding the selected topics onto his previous query [21].

The approach we used in this research for topic hierarchy construction is the Hierarchical Agglomerative Clustering and Partitioning algorithm (HAC+P) recently proposed for text documents [22]. This algorithm is performed on-line in real time on the retrieved spoken documents. It consists of two phases: an HAC-based clustering to construct a binary-tree hierarchy and a partitioning (P) algorithm to transform the binary-tree hierarchy to a balanced and comprehensive m-ary hierarchy, where m can be different integers at different splitting nodes. The first phase of HAC algorithm is kind of standard, based on the similarity between two clusters C_i and C_j and is performed bottom-up, while the second phase of partitioning is top-down. In this second phase, the binary-tree is partitioned into several sub-hierarchies first, and then this procedure is applied

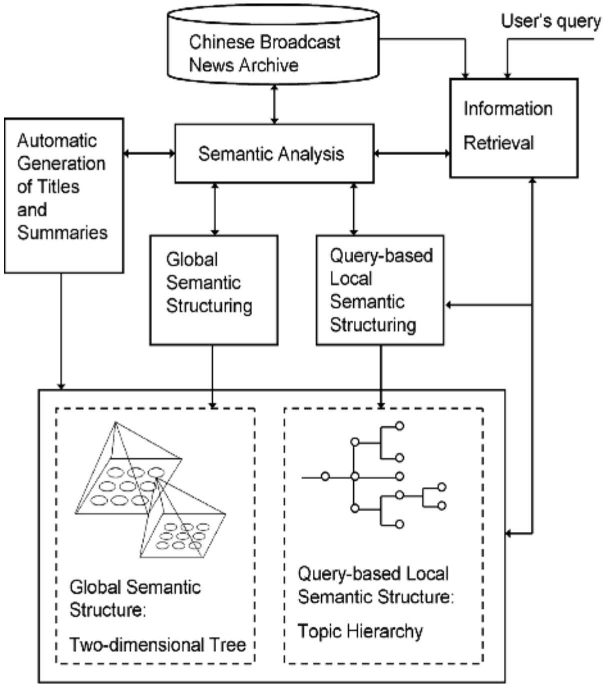

Chinese Broadcast News Archive
User's query
Information Retrieval
Automatic Generation of Titles and Summaries
Semantic Analysis
Global Semantic Structuring
Query-based Local Semantic Structuring
Global Semantic Structure:
Two-dimensional Tree
Query-based Local Semantic Structure:
Topic Hierarchy

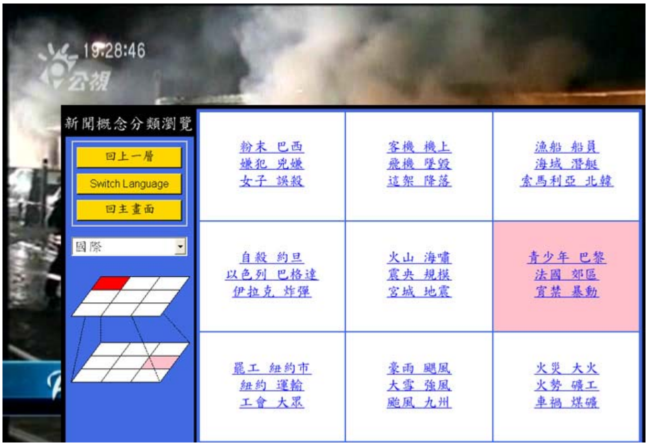
Fig. 1. The block diagram of the initial prototype system

recursively to each sub-hierarchy. The point is that in each partitioning procedure the best level at which the binary-tree hierarchy should be cut in order to create the best set of sub-hierarchies has to be determined based on the balance of two parameters: the cluster set quality and the number preference score [21].

3 An Initial Prototype System

An initial prototype system has been successfully developed. The broadcast news are taken as the example spoken/multi-media documents. The broadcast news archive to be summarized includes two sets, all in Mandarin Chinese. The first has roughly 85 hours of about 5,000 news stories, recorded from radio/TV stations in Taipei from Feb. 2002 to May 2003. No video signals were kept with them. The character and syllable error rates of 14.29% and 8.91% respectively were achieved in the transcriptions. The second set has roughly 25 hours of about 800 news stories, all including the video signal parts, recorded from a TV station in Taipei from Oct. to Dec. 2005. The character and syllable error rates for this set is 20.92% and 13.90%.

For those news stories with video signals, the video signals were also summarized using video technologies, for example, video frames for human faces, with moving objects and scene changes are more important, and the length of the video summary is based on the length of the speech summary. For the global



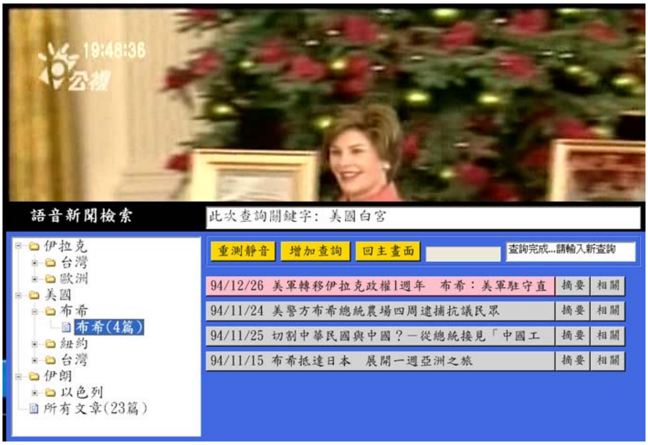
粉末 巴西 嫌犯 兇嫌 女子 誤殺	客機 機上 飛機 墜毀 這架 降落	漁船 船員 海域 潛艇 索馬利亞 北韓
自殺 約旦 以色列 巴格達 伊拉克 炸彈	火山 海嘯 震央 規模 宮城 地震	青少年 巴黎 法國 郊區 宵禁 暴動
罷工 紐約市 紐約 運輸 工會 大眾	豪雨 颶風 大雪 強風 颱風 九州	火災 大火 火勢 礦工 車禍 煤礦

Fig. 2. A 3x3 map on the second layer expanded from a cluster on the first layer of the global semantic structure for world news

semantic structure, a total of six two-dimensional tree structures were obtained for six categories of news stories, e.g. world news, business news, sports news, A 3x3 small map on the second layer of the tree for world news overlaid with the video signal is shown in Fig. 2. This is a map expanded from a cluster in the first layer covering all disasters happening worldwide. As can be found that on this map one small cluster is for airplane crash (墜機) and similar, one for earthquake (地震) and similar, one for hurricane (颶風) and similar, and so on. All news stories belonging to each node of the two-dimensional tree are listed under the node by their automatically generated titles. The user can easily browse through the titles or click to view either the summaries or the complete stories. With this structure it is much more easier for the user to browse the news stories either top-down or bottom-up. For the query-based local semantic structuring, the topic hierarchy constructed in real-time from the news stories retrieved by a query, "White House of United States (美國白宮)," is shown on the left lower corner of Fig 3, in which the three topics on the first layer are respectively Iraq (伊拉克), US (美國) and Iran (伊朗), and one of the node in the second layer below US is President George Bush (布希). When the user clicks the node of President George Bush, the relevant news stories are listed on the right lower corner by their automatically generated titles. The user can then click the "summary" button to view the summary, or click the titles to view the complete stories. Such information are overlaid with the news retrieved with the highest score.

4 Performance Evaluation

The performance evaluation of some key technologies are briefly summarized here.



語音新聞檢索

此次查詢關鍵字: 美國白宮

查詢完成...請輸入新查詢

- 伊拉克
  - 台灣
  - 歐洲
- 美國
  - 布希
    - 布希(4篇)
  - 紐約
  - 台灣
- 伊朗
  - 以色列
- 所有文章(23篇)

94/12/26 美軍轉移伊拉克政權1週年　布希：美軍駐守直

94/11/24 美警方布希總統農場四周逮捕抗議民眾

94/11/25 切割中華民國與中國？—從總統接見「中國工

94/11/15 布希抵達日本　展開一週亞洲之旅

Fig. 3. The result of query-based local semantic structuring for a query of "White House of United States"

4.1 Performance Evaluation of Automatic Generation of Titles and Summaries

118 broadcast news stories recorded at Taipei were used as the test documents in the evaluation for title generation, compared to human-generated reference titles. The objective performance measures used were precision, recall, and F1 scores calculated from the number of identical terms in computer-generated and human-generated titles. In addition, five-level subjective human evaluation was also performed, where 5 is the best and 1 is the worst, with two different metrics, "Relevance" calibrating how the titles are related to the documents, and "Readability" indicating how the titles are readable. In subjective human evaluation, each subject was given the reference titles with reference scores for some reference documents. The results for the previously proposed statistical Translation (ST) approach [13], Adaptive K-nearest-neighbor (AKNN) approach [19] and the new approach proposed here are listed in table 1. It can be found that the proposed approach performs significantly better in all measures, except with slightly lower readability than AKNN.

The F-measure results of the proposed summarization approach using N-gram co-occurrence statistics (ROUGE-1,2,3) and the longest common subsequence (ROUGE-L) evaluated with ROUGE package [23] with respect to human-generated summaries are shown in table 2 for summarization ratios of 10% and 30%. Here listed are two different ways to perform the automatic

Table 1. Evaluation results for title generation

Approaches	Precision	Recall	F1	Relevance	Readability
ST	0.0783	0.1513	0.1032	3.615	1.874
AKNN	0.1315	0.1074	0.1183	3.594	4.615
Proposed	0.2176	0.2119	0.2147	4.102	4.332

Table 2. Evaluation results for automatic summary generation

	ROUGE-1		ROUGE-2		ROUGE-3		ROUGE-L	
Summarization Ratio	10%	30%	10%	30%	10%	30%	10%	30%
Significance Score	0.27	0.48	0.18	0.40	0.16	0.36	0.26	0.47
Proposed	0.36	0.54	0.30	0.47	0.29	0.44	0.36	0.53

summarization: the well known and very successful significance score [15,16], and the approach proposed in this research respectively. It can be found that the proposed approach is better in all scores.

4.2 Performance Evaluation of Global Semantic Structuring

The performance evaluation for the global semantic structuring was performed on the TDT-3 Chinese broadcast news corpus [20]. A total of 47 different topics have been manually defined, and each news story was assigned to one of the topics, or as "out of topic". These 47 classes of news stories with given topics were used as the reference for the evaluation. We define the "Between-class to within-class" distance ratio as in equation (3),

$$R = \bar{d}_B / \bar{d}_W, \tag{3}$$

where $\bar{d}_B$ is the average of the distances between the locations of the two clusters on the map for all pairs of news stories manually assigned to different topics, and $\bar{d}_w$ is the similar average, but over all pairs of news stories manually assigned to identical topics. So the ratio R in equation (3) tells how far away the news stories with different manually defined topics are separated on the map. Apparently, the higher values of R the better. On the other hand, for each news story d_i, the probability $P(T_k|d_i)$ for each latent topic $T_k, k = 1, 2, \ldots, K$, was given. Thus the total entropy for topic distribution for the whole document archive with respect to the organized topic clusters can be defined as:

$$H = -\sum_{i=1}^{N} \sum_{k=1}^{K} P(T_k|d_i) \log(P(T_k|d_i)), \tag{4}$$

where N is the total number of news stories used in the evaluation. Apparently, lower total entropy means the news stories have probability distributions

Table 3. Evaluation results for the global semantic structuring

	Choice of Terms	Distance Ratio (R)		Total Entropy (H)
		Proposed	SOM	
(a)	W	2.34	1.11	5135.62
(b)	S(2)	3.38	1.04	4637.71
(c)	C(2)	3.65	1.03	3489.21
(d)	S(2)+C(2)	3.78	1.02	4096.68

more focused on less topics. Table 3 lists the results of the performance measure for the proposed approach as compared to the well-known approach of Self-Organized Map (SOM) [24] for different choices of the "term" t_j used, i.e., W(words), S(2)(segments of two syllables), C(2)(segments of two characters), and combinations.

4.3 Performance Evaluation of Query-Based Local Semantic Structuring

The performance evaluation for the query-based local semantic structuring was performed using 20 queries to generate 20 topic hierarchies [21]. The average values of correctness (C) and coverage ratio (CR) were obtained with some manual efforts. The correctness (C) is the measure if all the key terms in the topic hierarchy is correctly located at the right node position. It can be evaluated by counting the number of key terms in the topic hierarchy which have to be moved manually to the right node position to produce a completely correct topic hierarchy. The coverage ratio (CR) is the percentage of the retrieved news stories which can be covered by the key terms in the topic hierarchy. On average a correctness (C) of 91% and a coverage ratio (CR) of 97% were obtained.

5 Conclusion

In this paper we presented an initial prototype system which performs multi-layered summerization for multi-media archives for the purposes of efficient browsing and retrieval. This includes (1) Automatic Generation of Titles and Summaries for each of the spoken documents, (2) Global Semantic Structuring of the spoken document archive and (3) Query-based Local Semantic Structuring for the subset of spoken documents relevant to user's query. Satisfactory performance for the system was obtained.

References

1. L.-S. Lee, S.-Y. Kong, Y.-C. Pan, Y.-S Fu, and Y.-T Huang, "Multi-layered summarization of spoken document archives by information extraction and semantic structuring," in *Interspeech*, 2006, To Appear.
2. L.-S. Lee and B. Chen, "Spoken document understanding and organization," IEEE *Signal Processing Magazine*, vol. 22, no. 5, Sept. 2005.
3. CMU Informedia Digital Video Library project [online]. Available : http://www.informedia.cs.cmu.edu/.
4. Multimedia Document Retrieval project at Cambridge University [online]. Available : http://mi.eng.cam.ac. uk/research/Projects/Multimedia Document Retrieval/.
5. D.R.H Miller, T. Leek, and R. Schwartz, "Speech and language technologies for audio indexing and retrieval," *Proc. IEEE*, vol. 88, no. 8, pp. 1338–1353, 2000.
6. S. Whittaker, J. Hirschberg, J. Choi, D. Hindle, F. Pereira, and A. Singhal, "Scan: Designing and evaluating user interface to support retrieval from speech archives," in *Proc. ACM SIGIR Conf. R&D in Information Retrieval*, 1999, pp. 26–33.

7. A. Merlino and M. Maybury, "An empirical study of the optimal presentation of multimedia summaries of broadcast news," in *Automated Text Summarization*, I. Mani and M. Maybury, Eds., pp. 391–401. Eds. Cambridge, MA:MIT Press, 1999.
8. SpeechBot Audio/Video Search at Hewlett-Packard (HP) Labs [online]. Available : http://www.speechbot.com/.
9. S. Furui, "Recent advances in spontaneous speech recognition and understanding," *Proc. ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pp. 1–6, 2003.
10. Columbia Newsblaster project at Columbia University [online]. Available : http://www1.cs.columbia.edu/nlp/newsblaster/.
11. T. Hofmann, "Probabilistic latent semantic analysis," *Uncertainty in Artificial Intelligence*, 1999.
12. R. Jin and A. Hauptmann, "Automatic title generation for spoken broadcase news," in *Proc. of HLT*, 2001, pp. 1–3.
13. M. Banko, V. Mittal, and M. Witbrock, "Headline generation based on statistical translation," in *Proc. of ACL*, 2000, pp. 318–325.
14. B. Dorr, D. Zajic, and R. Schwartz, "Hedge trimmer: A parse-and-trim approach to headline generation," in *Proc. of HLT-NAACL*, 2003, vol. 5, pp. 1–8.
15. S. Furui, T. Kikuchi, Y. Shinnaka, and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," *IEEE Trans. on Speech and Audio Processing*, vol. 12, no. 4, pp. 401–408, 2004.
16. M. Hirohata, Y. Shinnaka, K. Iwano, and S. Furui, "Sentence extraction-based presentation summarization techniques and evaluation metrics," in *Proc. ICASSP*, 2005, pp. SP–P16.14.
17. S.-Y. Kong and L.-S. Lee, "Improved spoken document summarization using probabilistic latent semantic analysis (plsa)," in *Proc. ICASSP*, 2006, To Appear.
18. C.-C. Wang, "Improved automatic generation of titles for spoken documents using various scoring techniques," M.S. thesis, National Taiwan Univerisity, 2006.
19. S.-C. Chen and L.-S. Lee, "Automatic title generation for chinese spoken documents using an adaptive k-nearest-neighbor approach"," in *Proc. European Conf. Speech Communication and Technology*, 2003, pp. 2813–2816.
20. T.-H. Li, M.-H. Lee, B. Chen, and L.-S. Lee, "Hierarchical topic organization and visual presentation of spoken documents using probabilistic latent semantic analysis (plsa) for efficient retrieval/browsing applications," in *Proc. European Conf. Speech Communication and Technology*, 2005, pp. 625–628.
21. Y.-C. Pan, C.-C. Wang, Hsieh Y.-C., T.-H. Lee, Y.-S. Lee, Y.-S. Fu, Y.-T. Huang, and L.-S. Lee, "A multi-modal dialogue system for information navigation and retrieval across spoken document archives with topic hierarchies," in *Proc. of ASRU*, 2005, pp. 375–380.
22. S.-L. Chuang and L.-F. Chien, "A pratical web-based approach to generating topic hierarchy for text segments"," in *ACM SIGIR*, 2004, pp. 127–136.
23. C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proc. of Workshop on Text Summarization Branches Out*, 2004, pp. 74–81.
24. T. Kohonen, S. Kaski, K. Lagus, J. Salojvi, J. Honkela, V. Paatero, and A. Saarela, "Self organization of a massive document collection," *IEEE Trans on Neural Networks*, vol. 11, no. 3, pp. 574–585, 2000.

Initial Experiments on Automatic Story Segmentation in Chinese Spoken Documents Using Lexical Cohesion of Extracted Named Entities

Devon Li, Wai-Kit Lo, and Helen Meng

Human-Computer Communications Laboratory,
Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong, Hong Kong
{ycli, wklo, hmmeng}@se.cuhk.edu.hk

Abstract. Story segmentation plays a critical role in spoken document processing. Spoken documents often come in a continuous audio stream without explicit boundaries related to stories or topics. It is important to be able to automatically segment these audio streams into coherent units. This work is an initial attempt to make use of informative lexical terms (or key terms) in recognition transcripts of Chinese spoken documents for story segmentation. This is because changes in the distribution of informative terms are generally associated with story changes and topic shifts. Our methods of information lexical term extraction include the extraction of POS-tagged nouns, as well as a named entity identifier that extracts Chinese person names, transliterated person names, location and organization names. We also adopted a lexical chaining approach that links up sentences that are lexically "coherent" with each other. This leads to the definition of a lexical chain score that is used for story boundary hypothesis. We conducted experiments on the recognition transcripts of the TDT2 Voice of America Mandarin speech corpus. We compared among several methods of story segmentation, including the use of pauses for story segmentation, the use of lexical chains of all lexical entries in the recognition transcripts, the use of lexical chains of nouns tagged by a part-of-speech tagger, as well as the use of lexical chains of extracted named entities. Lexical chains of informative terms, namely POS-tagged nouns and named entities were found to give comparable performance (F-measures of 0.71 and 0.73 respectively), which is superior to the use of all lexical entries (F-measure of 0.69).

Keywords: Story boundary detection, lexical cohesion, informative terms extraction, named entities.

1 Introduction

Story segmentation plays a critical role in spoken document processing. Spoken documents often come in a continuous audio stream (e.g. in news broadcasts) without explicit boundaries related to stories or units. It is important to be able to automatically segment these audio streams into coherent units. The segmentation process is non-trivial since the physical audio contents of a story boundary may be

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 693 – 703, 2006.

very diverse – it may be a silent pause, a short duration of music, a commercial break, etc. A simple approach for detecting story boundaries may be based on cue word matching, but the cue words may be specific to the television/radio program and its period. Changes in the cue words will present a need to alter the heuristics in the system. Previous approaches have used a combination of prosodic, lexical, semantic and structural cues for story segmentation. They include audio energy levels and their changes [1], timing and melody of speech [2], novel nouns appearing in a short look-ahead window [3], word repetitions, synonyms and other associations [4]. In particular, Stokes et al. [4] proposed the use of lexical chaining that does not depend on specific cue word entries. Hence the approach is robust towards changes in the program and time. Previous work was done mostly in English text or speech recognition transcripts of English audio. Limited results were presented for Chinese. This paper reports on our initial attempt in the development of an automatic story segmentation system based on recognition transcripts of Chinese news audio. The approach includes extraction of informative lexical terms, including nouns and named entities; followed by the insertion of "lexical chains" that connects repeated informative terms. These chains are then used in scoring sentences for story boundary hypothesis. Figure 1 depicts an overview of the task of audio extraction from an audio/video news program, the process of recognition transcription, the process of story boundary detection and the use of detected boundaries for story segmentation.

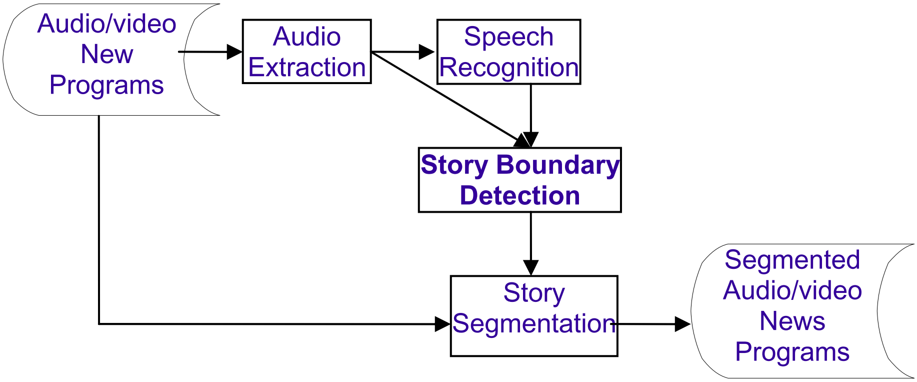


Fig. 1. Overview of the story segmentation task

2 Experimental Corpus and Evaluation

The experimental data is based on TDT2 Voice of American (VOA) Mandarin Speech corpus from February to June 1998 [5]. The VOA corpus contains radio news broadcasts in Mandarin and the corresponding recognition transcripts in GB-encoded Chinese characters. The recognition transcripts include pauses. Story boundaries are manually annotated in TDT2. Among 177 programs in TDT2, 54 programs (Feb to April) are used for training and parameter tunings and the rest of 123 programs (May to June) are used for evaluation. From the training set, there are 1,549 stories in the corpus where 1,106 are annotated as news stories. The remaining are classified as *miscellaneous*. Miscellaneous stories contain filler content during story transition (e.g. silence, music, both silence and music, etc.), a news summary, an advertisement or introductory and conclusive comments from the newscaster. For the evaluation set, there are 1,456 stories where 1,159 are annotated as news stories. According to the TDT2 evaluation plan version 3.7 [6], a hypothesized story boundary is considered "correct" if it is placed with 15 seconds of the manually defined reference boundary.

3 Overview of the Approach

Our approach for automatic story segmentation includes three phases: (i) informative lexical term extraction; (ii) lexical chaining and (iii) story boundary hypothesis. Details on each phase are presented in the following subsections.

3.1 First Phase – Extraction of Informative Lexical Terms (Nouns and Named Entities)

Our work on informative lexical terms extraction can draw heavily from previous work in the MUC (Message Understanding Conference) and MET (Multilingual Entity Task Evaluations) that focused on named entities (NE) [7]. Informative lexical terms refer to terms that carry useful content related to its story. Previous approaches have emphasized the use of *nouns* (see section 1). Other examples of informative terms are *named entities* include person names, location names, organization names, time and numeric expressions. For our experiments, we use an existing part-of-speech (POS) tagger [8] to perform part-of-speech tagging in Chinese. We extract the tagged nouns informative terms. We also develop a named entity extraction approach to extract Chinese person names, transliterated foreign person names, location names and organization names. It is well-known that the Chinese language presents a special research challenge for automatic lexical analysis due to the absence of an explicit word delimiter. A Chinese word may contain one or more characters and the same character set is used for both Chinese names and transliterated foreign names. Automatic lexical analysis of speech recognition transcripts faces the additional challenges of recognition errors and word segmentation errors. The latter arises because the speech recognizer's output is based on its (constrained) vocabulary, which is different from the open vocabulary in news audio. In view of this, we propose a lexicon-based approach and a purely data-driven approach for word tokenization followed by a series of NE filters to extract informative terms.

3.1.1 Word Tokenization

Figure 2 illustrates our word tokenization algorithm. Lexicon-based word tokenization involves a greedy algorithm that maximizes the length of the matching string as it references the CALLHOME lexicon. However, out-of-vocabulary (OOV) words absent from the lexicon will be tokenized as a series of singleton characters. Proper names are often OOV. For example:

贸易代表巴尔舍夫斯基
(translation: trade representative Barshefsky)

The character string is tokenized as:
［贸易］［代表］［巴］［尔］［舍］［夫］［斯］［基］

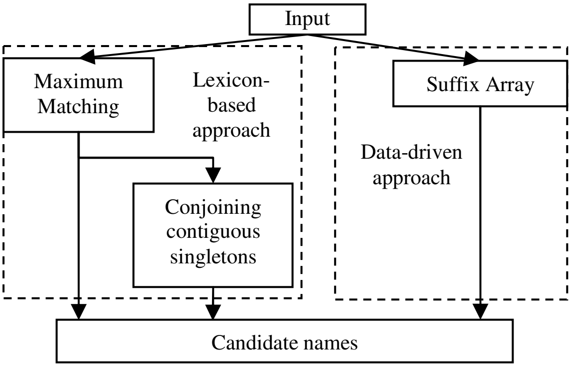


Fig. 2. Word tokenization using lexicon-based maximum matching as well as suffix array for generating candidate informative terms

Therefore, in order to salvage the OOV words that may potentially be a name, we conjoin the contiguous singletons to form a candidate name for names filtering in the subsequent module, i.e.:

［贸易］［代表］<巴尔舍夫斯基>
(translation: [trade] [representative] <Barshefsky>)

In other words, the lexicon-based approach leverages available lexical knowledge to extract OOV words that may be candidate names. However, since a given Chinese character sequence may have multiple possible word segmentations, the greedy algorithm may be biased by the lexicon and may miss out on other possible tokenization options. Hence, we supplement with a purely data-driven approach for word tokenization.

We used the suffix array structure [9] to extract the longest recurring string patterns in a radio program. The algorithm generates all substrings at all lengths within all sentences / utterances.[1] These substrings are then sorted. Only substrings that occur more than once and with lengths greater than one character are preserved, stop characters on either ends are removed and the resulting strings are treated as candidate names for subsequent names extraction. Analysis shows that the suffix array uncovers substrings such as “克林顿在” (*translation: “Clinton in”)* and “８万马克” (*translation*: *80,000 Deutsch Mark*), which may contain useful transliterated names.

[1] For the kth sentence with N_k characters ($C_1, C_2,\ldots,C_{Nk}$), possible output is $\{C_{ki}\ldots C_{kj},;\ \forall i=1$ to $N_k,\ \forall j=i$ to $N_k\}$.

3.1.2 Names Extraction

Four types of names will be extracted from the list of tokenized words: Chinese person names, transliterated person names, location names and organization names. For Chinese person name extraction, we apply two simple heuristics in this step: (a) the most common 100 surnames with reference to the surname list from [10], augmented with other surnames we found from the Web (i.e. 219 Chinese surnames in all); and (b) the popular Chinese names structure that consists of the surname (in one or two characters), followed by the given name (in one or two characters). Valid name structures include: SG, SGG, SSG and SSGG (where 'S' denotes a surname character, e.g. 陈; and 'G' denotes a given name character, e.g.红). Hence, in the Chinese name filtering procedure, a name candidate must be of two to four characters in length and must follow the pre-defined name structures in order to be qualified as a Chinese name.

Extraction continues with a transliterated name character bigram model that is trained on the MEI transliterated name list of 42,299 items [11]. We used Good-Turing discounting and backoff smoothing. By thresholding the normalized log probability at above -3, a 99% recall can be obtained from the training data. Log probability score are calculated for each word tokens, those scores above the threshold (i.e. -3) are extracted as transliterated person name.

A list of commonly used location and organization suffix characters are used to further extract word token which contains special suffix characters, e.g. "厅 署 中学 公司 城 池 村 道" (*translations: department, office, school, company, city, lake, village, road*). We also used a list of well-known location and organization names as basis for the extraction of known organization and location names.

3.2 Second Phase – Lexical Chaining

We extended the lexical chaining approach in [4] to Chinese with a focus on the repetition of informative lexical terms as an indicator of lexical cohesion. Lexical cohesion is represented as lexical chains that connect repeated occurrences of informative lexical terms. As mentioned previously, informative lexical terms include nouns or named entities. Other terms are deemed non-informative. If we observe a point in the story transcriptions where many existing lexical chains end and new lexical chains begin, then we consider it to be an indicator of a possible topic shift that is related to the occurrence of a story boundary.

More specifically, we group all contiguous words in the recognition transcripts of TDT2 into a "sentence". Hence the "sentences" are separated by pauses. Each sentence is labeled with an index number. Every informative lexical term in a sentence is tracked with regards to its occurrences in other sentences. This tracking process is conducted sequentially across the sentences that lie within a fixed window length, i.e. the window length for chain formation. Choice of values for this window length should reference the story length. In our training set, 94% of the stories range between 20 and 400 seconds in duration. At every 20 second we take a value and a total of twenty values are obtained. From these twenty values, the one that optimizes the training performance is selected. A lexical chain is inserted if the current sentence contains an informative lexical term that has occurred in the previous sentence. We define a "start chain" (i.e. starting lexical chain) to occur where a lexical term is chained to the following sentences but not the preceding sentences. Conversely, we define an "end chain (i.e. ending lexical chain) to occur where a lexical term is chained to the preceding sentences but not the following sentences. This is illustrated in Figure 3.

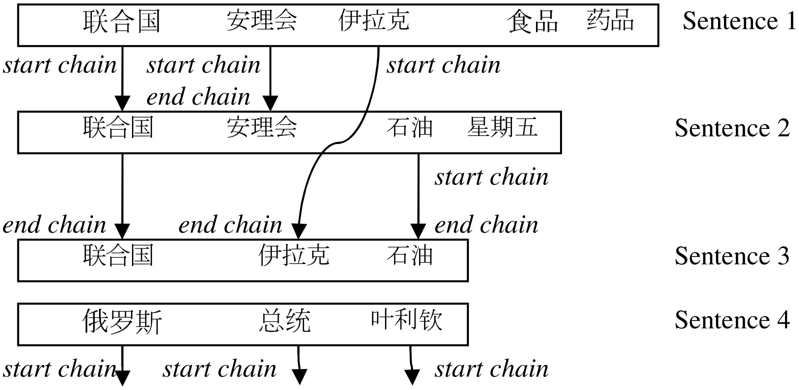


Fig. 3. Illustration of lexical chaining, Sentence 1 contains three start chains (联合国, 安理会, 伊拉克); Sentence 2 contains one end chain (安理会) as well as one start chain (石油); Sentence 3 contains three end chains (联合国, 伊拉克, 石油) and Sentence 4 contains three start chains (俄罗斯, 总统, 叶利钦)

3.3 Third Phase – Story Boundary Hypothesis

After all lexical chains are established, we assign a "chain score" to every sentence, defined as:

$$chainScore_{sent\_i} = num\_start\_chain_{sent\_i} + num\_end\_chain_{sent\_i-1} \quad \text{(Equation 1)}$$

Based on Equation (1), the chain scores for sentences 1 to 4 in Figure 3 should be 3, 1, 1 and 6 respectively. A sentence with a high chain score generally has a high number of start chains and its preceding sentence contains a high number of end chains. This should be an indication of a likely occurrence of a story boundary.

In this work, we take an *"over-hypothesize and filter"* approach to story boundary hypothesis. We will first hypothesize the occurrence of a story boundary if the chain score exceeds a tuned threshold as obtained from parameter tuning. We observed from the training data that story boundary existed at sentence with chain scores between one to nine. A tuned threshold can be obtained within this range. However, given that the evaluation criterion can tolerate offsets of 15 seconds for a story boundary (see Section 2), we also follow up with a *filtering mechanism* that selects the highest-scoring proposed boundary within a fixed window length. For example, in Figure 4 we see two sentences that lie 5.68 seconds apart but with two hypothesized boundaries. The filtering mechanism will remove the boundary at time 272.1 (with lower chain score=4) and keep the boundary at time 266.42 (with higher chain score =5). This parameter can be obtained by tuning in the development set

Before filtering hypothesized boundaries

```
<sen id=”25” time=”266.42” score=”5” boundary=”yes”>
        "苏联""格鲁吉亚""共和国""当局"</sen>
<sen id=”26” time=”272.1” score=”4” boundary=”yes”>
        "联合国""军队""观察员"</sen>
```

After filtering hypothesized boundaries

```
<sen id=”25” time=”266.42” score=”5” boundary=”yes”>
        "苏联""格鲁吉亚""共和国""当局"</sen>
<sen id=”26” time=”272.1” score=”4” boundary=”no”>
        "联合国""军队""观察员"</sen>
```

Fig. 4. Illustration of the filtering mechanism for hypothesized story boundaries. We take an “over-hypothesize and filter” approach to story boundary detection. Sentences with a chain score exceeding a trained threshold will be hypothesized with a story boundary. Given that the evaluation criterion can tolerate boundary offsets up to 15 seconds, our filtering mechanism uses a fixed window length within which only the highest-scoring boundary is preserved.

where given that there is a 15 seconds offsets during evaluation, a value smaller than 30 seconds will be a reasonable candidate for this parameter.

4 Experimental Results

We have a series of comparative experiments on automatic story segmentation. The various experimental setups are:

1. **Baseline performance using pauses:** The first baseline segments stories based on the occurrence of pauses. A story boundary is hypothesized whenever a pause occurs in the recognition transcripts. This is a very aggressive baseline segmenter, since pauses may also result from breath breaks, turn-taking in a dialog, etc. which do not correspond to a story boundary.
2. **Baseline performance using all lexical terms:** The second baseline includes all lexical terms found in the recognition transcripts for lexical chaining. Hence the vocabulary used for lexical chaining is identical to that of the speech recognizer.
3. **Performance of lexical chaining with POS-tagged nouns:** An existing POS tagger [8] which is trained on another text corpus is used for tagging nouns (including locations). These are categorized as “informative lexical terms” and only such terms are used for lexical chaining and subsequent story boundary hypothesis.
4. **Performance of lexical chaining of extracted named entities:** In this setup, informative lexical terms are defined as extracted named entities, including Chinese personal names, transliterated personal names, location names and organization names. The method of extraction is described in Section 3.1. The extracted named entities are used in the lexical chaining experiments.

Table 1 shows the tuned values for each parameter from the training corpus. The window length for chain formation is consistent across all units and 80 is actually closed to the average story length (100 seconds) in the training corpus. The window length for boundary removal also consistent across all units at 25 seconds while the chain score differ with each other where the chain score value is in proportional to the number of terms available during chain formation. From the training corpus, the best performance is obtained by using named entities for chaining where it also achieved the best precision among other chaining units.

Table 1. shows tuned parameters for each lexical chaining unit in the training data set as well as their corresponding performance on story boundary detection

	All lexical terms	**POS-tagged nouns**	**Named entities**
Window length for chain formation (seconds)	80	80	80
Chain score threshold	4	3	2
Window length for boundary removal (seconds)	25	25	25
Number of terms	191,371	115,110	43,417
Precision(P)	0.55	0.58	0.63
Recall (R)	0.64	0.69	0.66
F-measure (F)	0.59	0.63	0.64

We applied the trained thresholds to the evaluation corpus and results are shown in Table 2 and Figure 5. Total number of terms for each chain unit in the evaluation corpus is 67,380, 43,051 and 18,872 respectively.

Table 2. Performance on story boundary detection based on (i) use of pauses in recognition transcripts; (ii) lexical chaining of all vocabulary items in recognition transcripts; (iii) lexical chaining of POS-tagged nouns; (iv) lexical chaining of extracted named entities

	Pauses only	**All lexical terms**	**POS-tagged nouns**	**Named entities**
Precision(P)	0.04	0.86	0.87	0.88
Recall (R)	1	0.57	0.63	0.59
F-meas. (F)	0.08	0.69	0.73	0.71

It can be observed from Figure 5 that story segmentation based on pauses produces very high recall (R=1) but very low precision (P=0.04), leading to an F-measure of 0.08. This is because there are over 14,700 pause segments in the corpus but only 1,159 correspond to story boundaries. The filtering mechanism removes a fraction of false alarms in story boundary hypotheses occurring within a 25-second window.

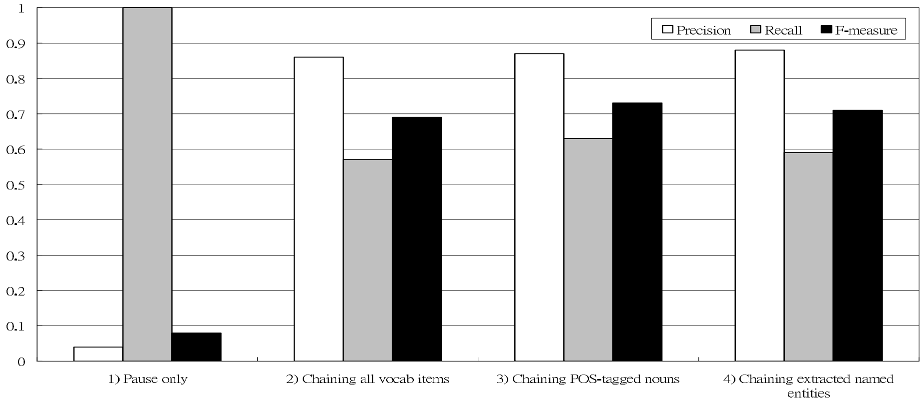


Fig. 5. Performance on story boundary detection based on (i) use of pauses in recognition transcripts; (ii) lexical chaining of all vocabulary items in recognition transcripts; (iii) lexical chaining of POS-tagged nouns; (iv) lexical chaining of extracted named entities

As we migrated to the use of lexical chaining of all vocabulary items in the recognition transcripts, performance values are P=0.86, R=0.57 and the overall F-measure improved to 0.69. The lexical chains offer lexical constraints for story segmentation. Lost recall is generally due to having too few lexical chains, causing the chain score to fall below the threshold, thereby missing the hypothesis of a story boundary. For example, one of the boundary sentences was “在 危地马拉” *(translation: in Guatemala),* where there is only one lexical chain, leading to a missed story boundary.

The use of POS-tagged nouns attains the performance of P=0.87, R=0.63 and further improved the F-measure to 0.73. We believe that these selected informative terms offers more focus in ascertaining lexical coherence. For example, the sentence “一个 发电厂 一个 核反应堆” (*translation: one power plant, one nuclear reactor*) contains three terms “"一个""发电厂""核反应堆"”. Two of these are tagged as nouns, i.e. “"发电厂", "核反应堆"”. The term “一个” is rather general and is generally not significant in the determination of lexical cohesion. It may even be possible for such terms to give rise to insignificant lexical chains that generate inaccurate story boundaries. This is an illustration of the possible benefits of using POS-tagged nouns for lexical chaining.

The use of extracted named entities gave the performance of P=0.88, R=0.59 and the F-measure of 0.71, which suggests that these are comparable with POS-tagged nouns for ascertaining lexical cohesion for story segmentation, with slightly better precision and slightly lower F-measure. In our analysis, we found that named entities are often more descriptive of lexical cohesion than general nouns (hence achieving better precision). For example, the sentence with the named entities,"美国 中东 特使 罗斯 星期一 以色列" (*translation: US, Middle East, special envoy, Roth, Monday, Israel*) contains five nouns “美国 中东 罗斯 星期一 以色列”. The term “星期一” (Monday) was lexically chained with a preceding sentence, which suggests lexical cohesion of this sentence with the preceding sentences. The remaining four terms were chained with following sentences which suggests lexical cohesion with following sentences and thereby outweighing the effect of the term “星期一”. Since named entities generally contain more distinctive information for describing lexical cohesion, they provide a better precision value for story segmentation. On the other hand, when compared with POS-tagged nouns, named entities achieve a lower recall for in both the training and evaluation corpora. This may be related to the use of 18.872 unique named entities in the data set, as compared with 43,051 POS-tagged nouns.

7 Conclusions and Future Work

This paper presents our initial experiments in automatic story segmentation of recognition transcripts of Chinese spoken documents. This is an important problem since spoken documents often come in a continuous audio stream without explicit boundaries that indicate the transition from one story (or topic) to another. Our approach consists of three phases:

- Automatic term extraction that includes lexicon-based maximum matching for word tokenization, followed by POS tagging and nouns extraction. We also develop a named entity extraction approach, involving lexicon-based maximum matching to uncover out-of-vocabulary words as singleton characters, together with purely data-driven suffix array approach that identify recurring strings. These extracted terms are then passed through a series of filters for Chinese names, transliterated names, location and organization names.
- A lexical chaining algorithm that connects repeated informative lexical terms as an indication of lexical cohesion among sentences. Story boundaries tend to occur where many existing lexical chains end and new lexical chains begin.
- A story boundary hypothesis component that adopts an “over-hypothesize and filter” paradigm – the lexical chain score (based on the total number of ending and starting lexical chains) of each sentence is compared with a trained threshold, above which a story boundary will be proposed. This is followed by a filtering mechanism that checks whether multiple boundaries are hypothesized within a small time window (25 seconds), upon which only the highest-scoring boundary hypothesis is preserved.

We conducted story segmentation experiments based on TDT2 Voice of America Mandarin news data. We observe increasing F-measures in story segmentation performance as we migrate from using only pauses for story segmentation; using all vocabulary items in the recognition transcripts with lexical chaining; using informative terms with lexical chaining. These results suggest that named entities serve well as informative lexical terms that can effectively describe lexical cohesion for automatic story segmentation. Future work will incorporate the use of both POS-tagged nouns and named entities, synonyms and other word associates in HowNet [12] for lexical chaining; as well as the incorporation of other prosodic features, e.g. fundamental frequencies for story segmentation.

Acknowledgments

This research is partially supported by the CUHK Shun Hing Institute of Advanced Engineering and is affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

References

1. Greiff, W., Hurwitz, L., Merlino, A. MITRE TDT-3 Segmentation System, TDT Evaluation System Summary, 1999
2. Shriberg, E., Stolcke, A., Hakkani-Tur, D. & Tur, G., Prosody-Based Automatic Segmentation of Speech into Sentences and Topics, *Speech Communication* 32(1-2), 127-154, September 2000
3. Franz, M., McCarley, J.S., Ward T., Zhu, W.J., Segmentation and Detection at IBM: Hybrid Statistical Models and Two-tiered Clustering. TDT Evaluation System Summary, 1999
4. Stokes, N., Carthy, J., Smeaton, A.. SeLeCT: A Lexical Cohesion based News Story Segmentation System. In the Journal of AI Communications, Vol. 17, No. 1, pp. 3-12, March 2004.
5. TDT2 Main page, http://projects.ldc.upenn.edu/TDT2/
6. TDT2 Evaluation Plan 1998, v 3.7. http://www.nist.gov/speech/tests/tdt/tdt98/doc/tdt2.eval.plan.98.v3.7.ps
7. Palmer, D. and Ostendorf, M. Improved word confidence estimation using long range features, In EUROSPEECH-2001, 2117-2120.
8. Meng, H. and Ip, C. W., An Analytical Study of Transformational Tagging on Chinese Text, Proceedings of the 1999 ROCLING conference, August 1999.
9. Manber, U. and Myers, E.W. Suffix arrays: a new method for on-line string searches. SIAM Journal of Computing, 22(5), pp. 953-948., 1993
10. Yuan, 新百家姓出炉：李王张继续位列姓氏前三甲, http://news.sina.com.cn/c/2006-01-10/09097941017s.shtml, January 2006.
11. Meng, et al., Mandarin-English Information (MEI): Investigating Translingual Speech Retrieval, http://www.clsp.jhu.edu/ws2000/groups/mei/, 2000
12. HowNet, http://www.keenage.com

Some Improvements in Phrase-Based Statistical Machine Translation

Zhendong Yang, Wei Pang, Jinhua Du, Wei Wei, and Bo Xu

Hi-tech Innovation Center, Institute of Automation
Chinese Academy of Sciences, 100080 Beijing
{zdyang, wpang, jhdu, weiwei, xubo}@hitic.ia.ac.cn

Abstract. In statistical machine translation, many of the top-performing systems are phrase-based systems. This paper describes a phrase-based translation system and some improvements. We use more information to compute translation probability. The scaling factors of the log-linear models are estimated by the minimum error rate training that uses an evaluation criteria to balance BLEU and NIST scores. We extract phrase-template from initial phrases to deal with data sparseness and distortion problem through decoding. By re-ranking the n-best list of translations generated firstly, the system gets the final output. Some experiments concerned show that all these refinements are beneficial to get better results.

Keywords: phrase-based translation, minimum error rate training, phrase-template, re-scoring.

1 Introduction

Statistical machine translation is a promising approach to large vocabulary text translation. Inspired by the Candide system IBM developed in the early 1990s [1], many statistical machine translation systems have been proposed. From the word-based system initially, phrased-based and syntax-based translation systems have been developed [2][3].

We have proposed a phrase-based translation system [4]: In the system, we applies phrase-based translation model to capture the corresponding relationship between source and target language. A phrase-based decoder we developed employs a beam search algorithm, in which some target language words that have both high frequency of appearance and also fertility zero are introduced to make the result more reasonable. We improve the previously proposed tracing back algorithm to get the best path.

This paper shows some improvements of our system currently: Section 2 presents the architecture of our system. Section 3 describes how to extract phrase-template from initial phrases. Section 4 studies the approach to compute the translation probability and train the scaling factors of all the models used in the translation system. Our system uses some special information to re-score the n-best translations, this is outlined in Section 5. In Section 6, a series of experiments are presented. We analyze the results. We summarize our system in Section 7.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 704 – 711, 2006.

2 System Description

In statistical machine translation, we are given a source language (Chinese) sentence $c_1^J = c_1 \cdots c_j \cdots c_J$, the goal is to generate the target language (English) sentence $e_1^I = e_1 \cdots e_i \cdots e_I$ which maximize the posterior probability:

$$\begin{aligned} e_1^I &= \arg\max_{e_1^I} \{\Pr(e_1^I \mid c_1^J)\} \\ &= \arg\max_{e_1^I} \{\Pr(e_1^I)\Pr(c_1^J \mid e_1^I)\} \end{aligned} \tag{1}$$

Applying the maximum entropy framework [5], the conditional distribution $\Pr(e_1^I \mid c_1^J)$ can be modeled through suitable feature functions, our system is based on a log-linear model which extends the word-based IBM Model to phrase-based model. We obtain:

$$\begin{aligned} e_1^I &= \arg\max_{e_1^I} \{\Pr(e_1^I \mid c_1^J)\} \\ &= \arg\max_{e_1^I} \{\exp(\sum_{m=1}^{M} \lambda_m h_m(e_1^I, c_1^J))\} \ . \\ &= \arg\max_{e_1^I} \{\sum_{m=1}^{M} \lambda_m h_m(e_1^I, c_1^J)\} \end{aligned} \tag{2}$$

Our system uses some feature models to drive the translation process: translation model, language model, distortion model and future score model. The system exploits two search passes: the first is performed by a beam search [4] to obtain n-best results, the second is a re-scoring algorithm to get the final output. The process is illustrated as follows:

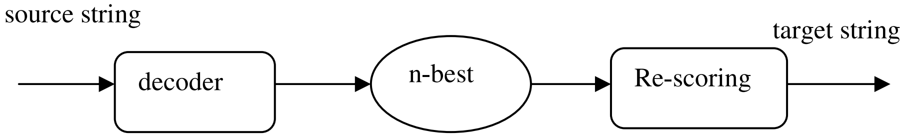


Fig. 1. The decoding illustration of our phrase-based translation system

3 Generalizing Phrases

The translation system often encounters data sparseness and distortion problem. Koehn et al. [6] find data sparseness takes over for long phrases. Many systems use very simple distortion model [6][7] to reorder phrases, which penalizes translations according to the jump distance instead of the syntax information. This model takes the risk of dealing with the long distance jump that is usually common between two languages with different expression habit. So we extract phrase-template from initial phrase pairs to alleviate these problems. The phrase are generalized in two ways:

If the phrase pairs include named entities such as named persons, locations and organization, or numeral, we replace them with some certain symbols in source and target sides. We call the generalized phrase of this type N_template. Some symbol examples are showed in table 1. Name entities are translated by separate rule-based module through searching process.

Table 1. Some symbol examples replacing name entities

type	Symbol in source side	Symbol in target side
named persons	PER_	per_
locations	LOC_	loc_
organizations	ORG_	org_
numeral	TIMP_	timp_

Besides this method, we generalize phrase pairs that don't include named entities as done inl [8], we call the rule generated X_template :

1. Initial phrase pairs are extracted from sentences similar to [6]. Every initial phrase pair is an X_template.
2. Aligned source-target small phrase included in initial phrases can be replaced by a nonterminal X. Then the phrase-template is extracted.
3. We only extract phrase template that has only one nonterminal and at least one terminal. This prevent from producing too many phrase-template.

The generalized phrases have some forms like that in table 2.

Because many phrase pairs may generate the same X_template. We select the highest translation probability of these phrase as that of X_template. But its real translation probability is the product of probability of X_template and the relevant nonterminal X. During decoding, only when the phrase can not be found in the phrase table, the corresponding phrase-template is used.

Table 2. The forms of phrase-template generated

Type	Initial phrase pairs (source # target)	Generalized phrase pairs (source # target)
N_tempate	9点到达 # get at 9	TIMP_到达 # get at timp_
	从 济南 到 武汉# from Jinan to Wuhan	从LOC_到LOC_ # from loc_ to loc_
X_template	与选民 建立 关系 # connect with the voters	与 X_1 建立 关系 # connect with X_1
	他的 朋友 之一 # one of his friends	X_1 之一 # one of X_1

4 Translation Model and Minimum Error Rate Training

There are several approaches to compute the phrase translation probability, Koehn estimate the probability distribution by relative frequency [6]:

$$h_1 = \frac{c(\tilde{c},\tilde{e})}{c(\tilde{c})} \quad . \tag{3}$$

or

$$h_2 = \frac{c(\tilde{c},\tilde{e})}{c(\tilde{e})} \quad . \tag{4}$$

Where $\tilde{e}$ and $\tilde{c}$ are English and corresponding Chinese phrase, and c(·) means the occurrence count in the training data. But if two phrase pairs have the same frequency, the probabilities have little discrimination. To get more discriminative probability, CMU calculate probabilities based on a statistical lexicon (such as IBM model 4) for the constituent words in the phrase, the formula is:

$$h_3 = \prod_i \sum_j p(c_i \mid e_j) \tag{5}$$

or inverse the formula:

$$h_4 = \prod_j \sum_i p(e_j \mid c_i) \ . \tag{6}$$

where c_i and e_j are the words that constitute phrase $\tilde{c}$ and $\tilde{e}$, $p(c_i \mid e_j)$ is the IBM model. This method has a drawback: If only one word of source phrase has no appropriate corresponding word in target phrase, the phrase translation probability will be small.

In order to offset the shortcoming of each method, we combine these four formulas to compute the phrase translation probability. The four formulas, distortion model, language model and future model are combined by log-linear form with a scaling factor each. The factors are estimated on the development data, by applying a minimum error training procedure [9]. It is an optimization problem, we use the simplex algorithm [10] to solve this problem. A key role of this training process is the evaluate metric. Firstly, we select BLEU as metric, we get a high BLEU score, but a low NIST score because the output sentences are short. Accordingly, we get a high NIST score at the cost of a significant deterioration of BLEU score when the NIST is used as the evaluate metric. A reasonable trade-off was final acquired using the metric:

100*BLEU + 5*NIST

The coefficient training process are introduced as follows:

1 Give every model scaling factor an initial value.

2 Use the current factor value to obtain the n-best candidate translations and corresponding features for each sentence through decoding. Merge the n-best lists across iterations.

3 Run the minimum error training to get the factor value of this iteration. If the value converges, the process stops, otherwise, goes to 2. The maximum of iteration is set as 10.

5 Re-scoring

The output sentence with the highest probability sometimes is not the best one compared with the reference translation. So we apply three additional feature functions to re-rank each of the 500 candidate translations for every input sentence:

- 2-gram target language model..
- 4-grams target language model
- Question feature, that is, if the input sentence ends with a question punctuation, we alleviate the penalizing on distortion.
- Name entity feature, i.e. if the number of name entity of output sentence is equal to that of source sentence, a binary feature is triggered to favor this translation.

We use the SRI Language modeling Toolkit to train language model. These features can be used respectively or combinatorially. We first get the top 500 candidates for every input sentence, then re-rank them to obtain the final output. The experiments are introduced in Section 6.

6 Experiments

We carry a number of experiments on 2005 863 Chinese-to-English MT evaluation tasks of China. 870,000 sentence pairs are used as training data to train the translation and language model. 500 Chinese sentences with about 4 reference translation sentences each are used as development data, we use the development data to optimize the model scaling factors. About 450 sentences are reserved for testing all the experiments. All these data are from the 2005 863 MT evaluation data. These sentences are about tour and daily life with the length of 5-20 words.

First we do experiments on the test data to check the role of the phrase-template. The experiments are made without training the model scaling factor. The results are shown in figure2. Where No_template denotes no phrase-template used, and +_template denotes adding phrase-template. We can see with the phrase-template added, the BLEU score goes up from 0.182 to 0.197, NIST score increases from 4.77 to 5.86. This experiment shows the phrase-templates play a positive role because they partly remedy the data sparseness and distortion problem. So we train the model factor by minimum-error-rate training with phrase-template added. The results are showed in figure 3 and table 3.

We make minimum error rate training on development set. We translate the sentence of test data to check the effect of the training, the results are showed in table 3. From figure 3, we can see the BLEU score' changing trend with the total number of translation candidates. The NIST score changes like this situation. The training procedure is iterated until the n-best list remains stable. In our experiment, about 9 iterations are needed for convergence. The final values of each model' scaling factor are showed in table 3. The BLEU score increases 0.015 from 0.197 to 0.212, and the NIST score goes up from 5.86 to 6.22.

Finally, we do experiments on the test set for re-ranking. Table 4 shows the single contribution of the 4 feature functions used. Almost all the features enhance the performance except question feature, this is because the phrase-template has partly resolve reordering phrases. This feature breaks the balance of all the models used. The result from 4-gram feature is superior to other methods, this indicates the n-gram feature provides a significant role on fidelity and fluency of the translation. Combining these methods always leads to some improvement.

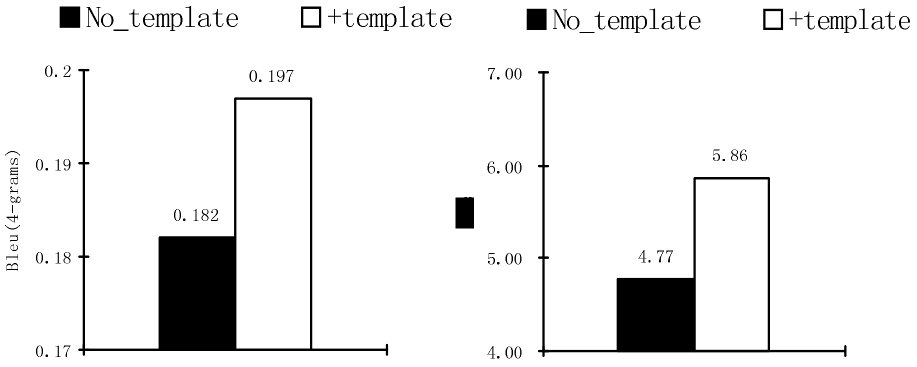


Fig. 2. The role of the phrase-template

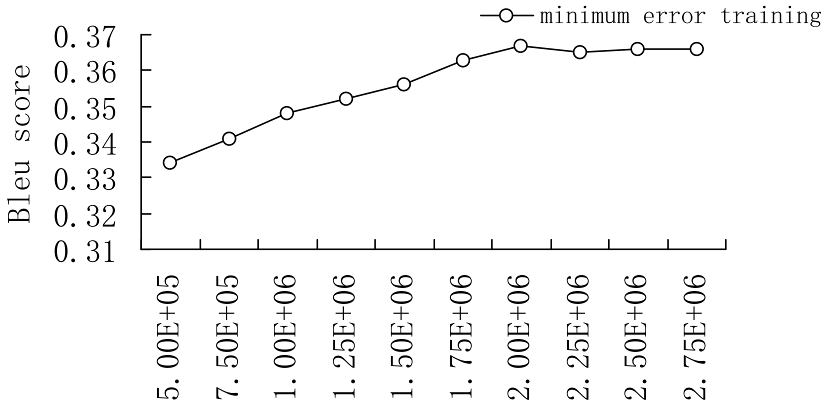


Fig. 3. Blue score as a function of the total number of generated translation candidates

Table 3. The final model factors and the BLEU and NIST score results when translating the test data. $\lambda_1 \sim \lambda_4$ mean the translation model factor, λ_{Lm}, λ_{dis} and λ_{fut} mean the scaling factor of language model, distortion modle and future score model respectively

λ_1	λ_2	λ_3	λ_4	λ_{Lm}	λ_{dis}	λ_{fut}
1.0628	0.0031	2.4476	0.7702	1.3992	0.1185	1.6576
BLEU (4-gram)		0.212				
NIST		6.22				

Table 4. The effect of each feature functions in re-scoring step on the test data

System	BLEU(4-gram)	NIST
baseline	0.212	6.22
Question feature	0.202	5.92
2-grams LM	0.213	6.31
4-grams LM	0.221	6.64
Name entity feature	0.216	6.52
All features	0.224	6.83

7 Conclusion

In summary, this paper shows some improvements to our phrase-based translation system. We use phrase-template to alleviate data sparseness and reorder the phrases during translation. The translation model is refined, and the scaling factors of the all the models are estimated by minimum error rate training. Instead of output the translation with the highest probability, we re-score the n-best lists to get the final translation. All these efforts are effective to our system.

Although we used some formal syntax to generalize the phrases, how to combine syntax with phrase is our important work next. We will do some studies about parsing to improve our phrase-based system next step.

References

1. Peter F. Brown , Stephen A. Della Pietra, Vincent J. Della Pietra, and Pobert L. Mercer. The Mathematics of Statistical Machine Translation: Parameter Estimation. Computational Linguistics, vol. 19, no. 2, (1993), pp. 263-311,.
2. Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venugopal, Bing Zhao, Alex Waibel . The CMU Statistical Machine Translation System. In proceedings of the Ninth Machine Translation Summit.(2003).
3. Yamada, K. and Knight. A Syntax-based Statistical Translation Model. In Proc. of the 39th Annual Meeting of ACL, (2001).
4. Zhendong Yang, ZhenBiao Chen, Wei Pang, Wei Wei, Bo Xu, The CASIA Phrase-Based Machine Translation System, IEEE NLPKE'05, Wuhan, China, (2005),pp.416-419.

5. A Berger, S. Della Pietra, and .Della Pietra, “A Maximum Entropy Approach to Natural Language Processing”, Computational Linguistics, vol. 22, no.1, (1996), pp 39-71.
6. Koehn, P. ,Och, F. J., and Marcu , D. Statistical Phrase-Based Translation. In Proceedings of the Joint Conference on Human Language Technologies and the Annual Meeting of the North American Chapter of the Association of Computational Linguistics. (2003).
7. Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. Computational Linguistics, 30:417-449.
8. David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In Proceedings of ACL 2005.pages 263-270.
9. Franz Josef OCH. Minimum Error Rate Training in statistical machine translation . In Proc. of the 41th Annual Meeting of the Association of Computational Linguistics(ACL), Sapporo, Japan.2003
10. William H. PRESS, Saul TEUKOLSKY, William T. VETTERLING and Brian P. FLANNERY. 2002. Numerical Recipes in C++. Cambrige University Press, Cambridge, UK.

Automatic Spoken Language Translation Template Acquisition Based on Boosting Structure Extraction and Alignment

Rile Hu and Xia Wang

Nokia Research Center, Nokia House 1, No. 11, He Ping Li Dong Jie,
Beijing, 100013
{ext-rile.hu, xia.s.wang}@nokia.com

Abstract. In this paper, we propose a new approach for acquiring translation templates automatically from unannotated bilingual spoken language corpora. Two basic algorithms are adopted: a grammar induction algorithm, and an alignment algorithm using Bracketing Transduction Grammar. The approach is unsupervised, statistical, data-driven, and employs no parsing procedure. The acquisition procedure consists of two steps. First, semantic groups and phrase structure groups are extracted from both the source language and the target language through a boosting procedure, in which a synonym dictionary is used to generate the seed groups of the semantic groups. Second, an alignment algorithm based on Bracketing Transduction Grammar aligns the phrase structure groups. The aligned phrase structure groups are post-processed, yielding translation templates. Preliminary experimental results show that the algorithm is effective.

Keywords: Spoken language processing, machine translation, translation template extraction, structure extraction and alignment.

1 Introduction

With the development of corpus processing technology, more and more bilingual corpora are becoming available for knowledge acquisition in machine translation (MT) and many other natural language processing tasks. Translation templates provide one especially useful kind of knowledge for MT systems. At the same time, phrasal translation examples are an essential resource for many MT and machine-assisted translation architectures. In this paper, we bring the need and the resource together. We present a new approach for acquiring translation templates automatically from a sentence-aligned parallel English-Chinese corpus through structure extraction and alignment.

In some early-built example-based machine translation systems, the translation templates are extracted manually from the corpus. For example, [1] manually encodes translation rules in this way. Similarly, [2] has also proposed an example-based system which employs manually-built matching expressions as translation templates.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 712 – 723, 2006.

However, as the size of corpus grows, the manual process of template extraction becomes increasingly difficult and error-prone.

Some methods for automatically acquiring translation templates have also been proposed. For instance, in [3], the analogical models are adopted for learning translation templates from bilingual parallel corpus. Templates are obtained by grouping the similar translation examples and replacing the variances with variables. However, such methods rely on a very large bilingual parallel corpus, which contains many similar instances. By contrast, some other methods for template acquisition are instead based on structure alignment [4][5]. Those approaches follow a procedure which may be termed "parse-parse-match" [6], i.e. each language in the parallel corpus is first parsed separately by using monolingual grammars, then the corresponding constituents are matched using some heuristic procedures. The performance of those methods is highly dependent on the parsers of the source and target languages.. In a similar vein, [7] has proposed a scheme based on bilingual language modeling: bilingual sentence pairs are first aligned with respect to syntactic structure by combining a parser with a statistical bilingual language model. The translation templates are produced from the alignment results. This scheme, likewise, needs a high-performance parser, as well as the part-of-speech tagging systems for both the source and the target language.

And some other statistical methods are also proposed to perform the task of translation template acquisition. [6] introduced the Bracketing Transduction Grammar (BTG). It uses no language specific syntactic grammar, and employs a maximum-likelihood parser to select the parse tree that best satisfies the combined lexical translation preference. This method achieves encouraging results for bilingual bracketing using a word-translation lexicon alone. [8] proposed the alignment template translation model. It explicitly takes shallow phrase structures into account, using two different alignment levels: a phrase level alignment between phrases and a word level alignment between single words. This method is capable of completely automatic learning by using a bilingual training corpus and can achieve better translation results on a limited-domain task than other example-based or rule-based translation systems. And [9] propose a new alignment model based on shallow phrase structures, and the structures can be automatically acquired from parallel corpus.

In this paper, we propose a statistical, data-driven approach which acquires translation templates from unannotated bilingual corpora based on the bilingual grammar induction and BTG.

2 Our Motivations

The translation template acquisition based on structural alignment is a popular method in the area of statistical machine translation. Considerable research has been carried out on this topic. We focus here on the methods based on unsupervised machine learning;

and propose a translation template acquisition method based on statistical phrase structure extraction and alignment.

The main ideas of our approach to translation template acquisition are shown in Fig. 1:

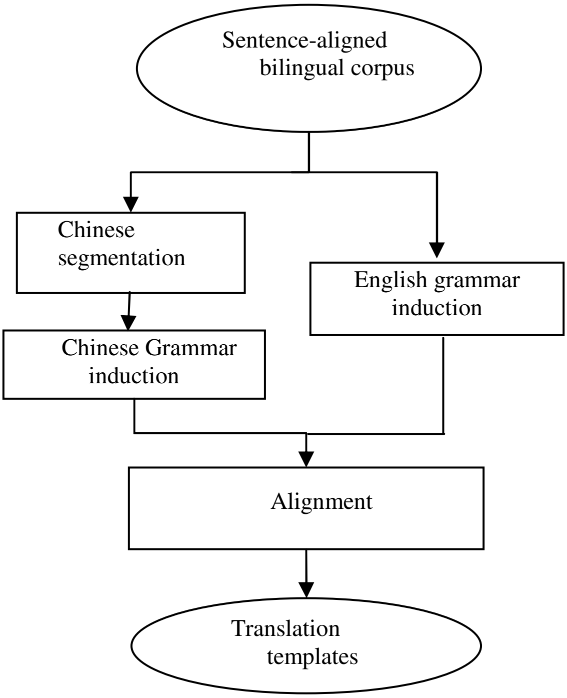

Sentence-aligned bilingual corpus
Chinese segmentation
English grammar induction
Chinese Grammar induction
Alignment
Translation templates

Fig. 1. Architecture of the proposed translation template acquisition system

The input of our approach is the sentence-aligned bilingual corpus. Here, an English-Chinese bilingual corpus is used. The Chinese sentences are first segmented; then the grammar induction procedure is performed on both English and Chinese sentences. Next, the semantic groups (labeled SCi) and phrasal groups (labeled PCi) are obtained from the corpus for both English and Chinese. Finally, the phrase structures of the languages are aligned, using a modified BTG. These aligned phrase structures are post-processed to create the translation templates, which are the results of our approach.

We now give a simple example to explain how the translation templates are acquired from the unannotated corpora.

Suppose some SCi and PCi groups are obtained from the corpus, as shown in Table 1:

Table 1. Examples of the grammars acquired from the experimental corpus

Chinese part:

SCC10 → 单人间 | 双人间 | 标准间
PCC3 → 一 个
PCC8 → PCC3 SCC10
PCC12 → 我 想 预订
PCC20 → PCC12 PCC8

English part:

SCE5 → single | double | standard
PCE2 → want to
PCE4 → a SCE5 room
PCE8 →I PCE2 reserve
PCE14 → PCE8 PCE4

With the grammars shown in Table 1, the system aligns the phrase structures as follows:

[[I/我 [want/想 to/ε] reserve/预订] [a/一 ε/个 N* room/N]].

Where, to/ε means the Chinese word aligned with the word "to" is null, and

N=单人间⇔ N*=single;
N=双人间⇔ N*=double;
N=标准间⇔ N*=standard.

Thus we can obtain the following translation templates:

我 想 预订⇔ I want to reserve ①
一个N ⇔ a N* room ②

Here in ②, N and N* are shown above.

Here, ① and ② exemplify two kinds of translation templates in our approach. ① is a constant template, since all of its elements are constants. By contrast, ② contains at least one variable element, so we call such kind of temples the variable template.

3 Basic Algorithms

In this section, we provide a brief overview of our basic algorithms for grammar induction and alignment using BTG.

3.1 Grammar Induction Algorithm

This clustering method consists of two steps, spatial clustering and temporal clustering. In the clustering procedure, we consider entities as processing units. The entities include single words, semantic group labels, and the phrasal structure group labels obtained with the procedure of clustering. For example, in table 1, the single words 'single', 'want' and etc. are single-word-level entities; the semantic group labels SCE5 and SCC10 are the semantic-group-level entities, each entity containing a group of single words; and the phrasal structure group labels PCC8, PCE14 and etc. are the phrasal-structure-group-level entities, each entity containing a sequence of words or a sequence of entities. In the spatial clustering step, the entities which have similar left and right contexts are grouped together. These entities generally have similar semantics. In the temporal clustering step, the entities which frequently co-occur are clustered into groups. These entity groups tend to be commonly-used phrases.

In spatial clustering, the Kullback-Leibler distance is used to describe the similarity of the distributions of the local contexts of entities, where an entity's local context consists of the entity immediately before it and the entity immediately after it (1).

$$D(p_1 \| p_2) = \sum_{w_i \in V} p_1(w_i) \log \frac{p_1(w_i)}{p_2(w_i)} \tag{1}$$

Here, p_1 denotes the unigram distribution of the words which appear in the local context of the entity e_1; p_2 denotes the same distribution for entity e_2; and w_i denotes the word which appears in the local contexts of the entities e_1 and that of e_2, and V denotes the union of w_i.

In order to acquire a symmetric measure of the distance, or degree of difference, between two local context distributions, we use the divergence of the distributions, as shown in Formula (2):

$$Div(p_1, p_2) = D(p_1 \| p_2) + D(p_2 \| p_1) \tag{2}$$

Then the distance between two entities e_1 and e_2 is defined as in Formula (3):

$$Dist(e_1, e_2) = Div(p_1^{left}, p_2^{left}) + Div(p_1^{right}, p_2^{right}) \tag{3}$$

Distance between entities is thus the sum of the divergences of the distributions of the entities' left and right contexts.

In order to increase the clustering accuracy, we introduce the extended distance contexts into the measurement of distance between entities: we consider the words next to the entities' contexts, called extended contexts.

Finally, the distance between two entities is computed as the sum of the distance of the contexts and that of the extended contexts. Thus the distance between entities e_1 and e_2 can be described using Formula (4):

$$\begin{aligned} Dist^*(e_1, e_2) &= Div(p_1^{left}, p_2^{left}) + \tfrac{1}{2} Div_2(p_1^{left}, p_2^{left}) \\ &+ Div(p_1^{right}, p_2^{right}) + \tfrac{1}{2} Div_2(p_1^{right}, p_2^{right}) \end{aligned} \tag{4}$$

Here, the expression $Div_2(p_1,p_2)$ denotes the symmetric distance of the extended contexts of the two entities e_1 and e_2.

The maximally similar entities are gathered into a semantic group, labeled SCi. That is, we cluster the pairs of entities which have the minimal distance between them (as calculated with (4)).

Other measures which can be used to calculate the similarity between two entities have also been considered. We use feature vectors to describe the contexts of an entity, and these can be used to calculate the similarity between two entities. If an entity e appears in the context of another given entity, this relationship can be described using the expression *(posi, e)*, where *posi* has the value 'left' if *e* appears to the left side of the entity, or 'right' if *e* appears to the entity's right. The value of each feature is the frequency count of the feature in the corpus.

$(u_1,u_2,\ldots,u_n)$ and $(v_1,v_2,\ldots,v_n)$ denote the feature vectors for the entity *u* and *v*, *n* is the number of feature types extracted from the corpus, and *f(i)* is the *ith* feature.

Three other similarity measures are also used in the spatial clustering step, the Cosine Measure, Cosine of Pointwise Mutual Information, and Dice Co-efficient.

The Cosine Measure computes the cosine of two entities' feature vectors (5):

$$Cos(u,v) = \frac{\sum_{i=1}^{n} u_i \times v_i}{\sqrt{\sum_{i=1}^{n} u_i^2} \times \sqrt{\sum_{i=1}^{n} v_i^2}} \tag{5}$$

The pointwise mutual information (PMI) between a feature *f(i)* and an entity *u* measures the strength of the association between them, as defined in Formula (6):

$$PMI\,(f(i),u) = \log\left(\frac{P(f(i),u)}{P(f(i)) \times P(u)}\right) \tag{6}$$

Here, *P(f(i),u)* is the probability of *f(i)* co-occurring with *u*; *P(f(i))* is the probability of *f(i)* co-occurring with any entity; and *P(u)* is the probability of any feature co-occurring with *u*. For example, if all the features occur 1,000 times in the corpus, *f(i)* occurs 50 times, *f(i)* co-occurs with *u* 10 times, and *u* co-occurs with a feature 100 times, then *P(f(i),u)* = 10 / 1000 = 0.01 , *P(f(i))* = 50 / 1000 = 0.05, *P(u)* = 100 / 1000 = 0.1.

The Cosine of Pointwise Mutual Information (CosPMI) is defined in (7):

$$CosPMI\ \ (u,v) = \frac{\sum_{i=1}^{n} PMI\,(f(i),u) \times PMI\,(f(i),v)}{\sqrt{\sum_{i=1}^{n} PMI\,(f(i),u)^2} \times \sqrt{\sum_{i=1}^{n} PMI\,(f(i),v)^2}} \tag{7}$$

This formula computes the cosine between two entities' pointwise mutual information.

The Dice Co-efficient is defined in Formula (8). It is a simple measure of the difference between zero and non-zero frequency counts.

$$Dice\,(u,v) = \frac{2 \times \sum_{i=1}^{n} s(u_i) \times s(v_i)}{\sum_{i=1}^{n} s(u_i) + \sum_{i=1}^{n} s(v_i)} \tag{8}$$

Here, *s(x)*=1 if *x*>0 and *s(x)*=0 otherwise.

After the spatial clustering, we substitute a category label throughout the corpus for the words that have been grouped. Then the temporal clustering is computed.

In the temporal clustering step, the Mutual Information (MI) is used to describe the degree of co-occurrence of two entities e_1 and e_2 in the same sentence of the corpus, and it becomes the metric used for clustering. MI is defined in (9):

$$MI\,(e_1, e_2) = P(e_1, e_2)\log\frac{P(e_2 \mid e_1)}{P(e_2)} \tag{9}$$

The entities which have the highest MI are clustered into phrasal groups labeled PCi. Next, PC labels are substituted for these entity pairs. Then another iteration of spatial clustering can be started. This is a boosting procedure. After the application of the clustering algorithm, the semantic groups and phrase structure groups will be extracted from the corpus.

After each iteration of the clustering algorithm, more words are clustered into semantic groups and phrasal structure groups. The coverage of the clustering algorithm can be measured in terms of the percentage of words in the input corpus that are captured in the clustering groups. A stopping criterion (STC) is defined as the relative increment of the clustering coverage. For example, if the coverage after iteration is 80% and that of next iteration is 82%, then the STC between these two iterations is (82-80)/80=3.75%. When the STC is below 1%, the clustering algorithm will be stopped.

We now describe our grammar induction approach. Importantly, it can capture semantic and phrase structures from unannotated corpora.

The grammar induction algorithm is described in Fig. 2.

The input of the algorithm is either the English part or the Chinese part of the bilingual corpus.

Step1: If the distance measure is used, calculate the distance between each entity e1 and e2 in the corpus using Formula (4). If other similarity measures are used, calculate the similarity between each entity e1 and e2 using Formula (5), (7) or (8).
Step2: Group the N pairs which have the minimum distance or the maximum similarity into a semantic class.
Step3: Replace the entities in Step2 with their semantic class label SCi.
Step4: Use Formula (9) to calculate the MI between each entity e1 and e2 in the corpus.
Step5: Select the N pairs of entities with the highest MI to form the phrasal structure groups.
Step6: Replace the entities in Step5 with their phrasal structure class label PCi.
Step7: Calculate the STC. If the STC is lower than 1%, stop the procedure of the clustering algorithm, else go to Step 1.

The Output of the algorithm is a list of semantic groups and phrasal structure groups.

In the step of the spatial clustering, we introduce a synonym dictionary which is called “tong yi ci ci lin”. It is a Chinese synonym dictionary. We pick out the words both in the training corpus and in the synonym dictionary as the seed groups of the clustering. These groups are looked as the initial groups of the spatial clustering. And we also give out some manually built seed groups based on common sense, such as “Monday | Tuesday | …” in English and “星期一 | 星期二 | ……” in Chinese.

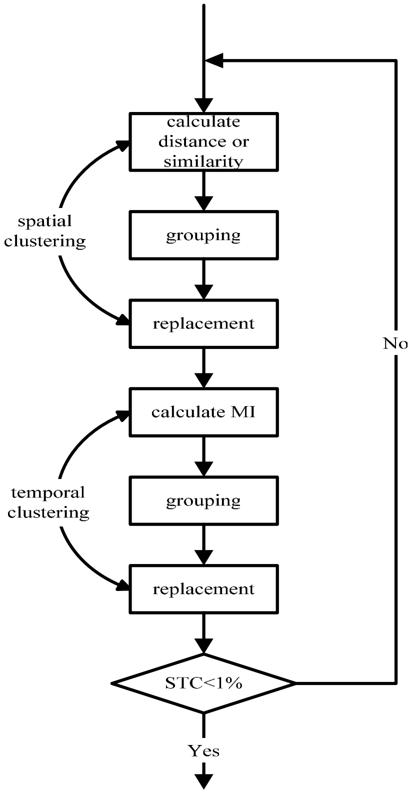


Fig. 2. Flow chart of the grammar induction algorithm

3.2 Alignment Using Bracketing Transduction Grammar

A bilingual model called Inversion Transduction Grammar (ITG), proposed by [6], parses bilingual sentence pairs simultaneously. As it is difficult to find suitable bilingual syntactic grammars for English and Chinese, we employ a simplified ITG called BTG.

The expressive characteristics of ITG grammars naturally constrain the space of possible matching in a highly appropriate fashion. As a result, BTG grammars achieve encouraging results for bilingual bracketing using a word-translation lexicon alone [6]. However, since no language specific syntactic knowledge is used in BTGs, the grammaticality of the output cannot be guaranteed [7].

Our main idea in the present work is to use phrase structure information acquired by the grammar induction algorithm as a boundary restriction in the BTG language model. When the constraint is incompatible with BTG, BTG is used as the default result. This procedure allows the alignment to continue regardless of some failures in the matching process. Then a dynamic programming algorithm is used to compute the maximally probable alignment of all possible phrase structures.

A constraint heuristic function $F_e(s,t)$ is defined to denote the English boundary constraint. Here, s denotes the beginning position of the phrase structure and t denotes its end position. Phrase structure matching can yield three cases: invalid match, exact match, and inside match. An invalid match occurs when the alignment conflicts with phrasal boundaries. Examples appear in (1,2), (3,4) and (4,5) etc. in the sample sentence below. (The constraint function is set at a minimum value 0.0001 to prevent selection of such matches when an alternate match is available.) An exact match means that the match falls exactly on the phrase boundaries, as in (2,3), (1,4) and (5,7) below. (When this condition is met, the function is set at a high value 10 for weighting.) Examples of inside matches are seen in (5,6) and (6,7) below. (The value of these functions is set to 1.)

Example:

[[I/1 [want/2 to/3] reserve/4][a/5 single/6 room/7]].

The Chinese constraint function $F_c(u,v)$ is defined similarly.

Now let the English input sentence be $e_1,\ldots,e_T$ and let the corresponding Chinese sentence be $c_1,\ldots,c_V$. As an abbreviation, we write $e_{s\ldots t}$ for the sequence of English words $e_{s+1}, e_{s+2},\ldots, e_t$; Similarly, we write $c_{u\ldots v}$ for the Chinese word sequence. Further, the expression $q = (s,t,u,v)$ identifies all possible matched structures, where the substrings $e_{s\ldots t}$ and $c_{u\ldots v}$ both derive from the node q. The local optimization function is shown in (10):

$$\delta(s,t,u,v) = maxP[q] \tag{10}$$

Equation (10) denotes the maximally probable alignment of the phrase structures. Then the best combination of the phrase structures has the probability $\delta(0,T,0,V)$.

To insert the English and Chinese constraints into the alignment procedure, we integrate the constraint functions $F_e(s,t)$ and $F_c(u,v)$ into the local optimization function. For this purpose, the function is split into three functions, as in formulas (11), (12) and (13) below.

$$\delta(s,t,u,v) = max[\delta^{[]}(s,t,u,v), \delta^{<>}(s,t,u,v)] \tag{11}$$

$$\delta^{[]}(s,t,u,v) = \max_{\substack{s \le S \le t \\ u \le U \le v \\ (S-s)(t-S)+(U-u)(v-U) \ne 0}} F_e(s,t)F_c(u,v)\delta_1\delta_2 \tag{12}$$

$$\delta^{<>}(s,t,u,v) = \max_{\substack{s \le S \le t \\ u \le U \le v \\ (S-s)(t-S)+(U-u)(v-U) \ne 0}} F_e(s,t)F_c(u,v)\delta_3\delta_4 \tag{13}$$

Here,

$$\delta_1 = \delta(s,S,u,U)$$

$$\delta_2 = \delta(S,t,U,v)$$

$$\delta_3 = \delta(s,S,U,v)$$

$$\delta_4 = \delta(S,t,u,U)$$

In (12) and (13), the condition *$(S-s)(t-S)+(U-u)(v-U)\neq 0$* specifies that only one of the language strings, not both, may be split into an empty string.

Other symbols in the algorithm are defined as follows: *$\theta(s,t,u,v)$*, *$\sigma(s,t,u,v)$* and *$\gamma(s,t,u,v)$* are the variables used to record the production direction, the spilt points in English, and the split points in Chinese, when *$\delta(s,t,u,v)$* is achieved. These variables are used to reconstruct the bilingual alignment tree in the final step. *$\lambda(s,t,u,v)=\lambda(q)$* is the non-terminal label of the node *q*. *LEFT(q)* is the left side of q, and *RIGHT(q)* is its right side.

The optimal bilingual parsing tree for a given sentence-pair is then computed using the dynamic programming (DP) algorithm [6] shown in Table 2:

Table 2. Alignment algorithm

1. Initialization

$\delta(t-1,t,v-1,v) = b(e_t / c_v) \quad 1 \le t \le T, \quad 1 \le v \le V$

$\delta(t-1,t,v,v) = b(e_t / \varepsilon) \quad 1 \le t \le T, \quad 1 \le v \le V$

$\delta(t,t,v-1,v) = b(\varepsilon / c_v) \quad 1 \le t \le T, \quad 1 \le v \le V$

2. Recursion

For all *s,t,u,v* which are restricted by

$0 \le s < t \le T, \quad 1 \le u < v \le V,$

$t - s + v - u > 2$

Calculate $\delta(s,t,u,v)$ using Formula (11), (12) and (13).

3. Reconstruction

Reconstruct and obtain the optimal result of the parsing tree.

4 Experiments

4.1 Training Set and Testing Set

In our experiment, 50,000 pairs of Chinese-to-English spoken parallel sentences and 1,000 monolingual Chinese sentences are randomly selected from the BTEC corpus as the training data and the testing data respectively. The sentences in the testing set are not concluded in the training set.

4.2 Experiment Results

The training set is used to extract the translation templates. The whole procedure of the translation templates extraction is carried on this set. The word alignment probabilities used in the step of phrase structure alignment are trained by the GIZA++ toolkit which performs statistical alignment (http://www.fjoch.com/GIZA++.html).

We have done four experiments, the first one is based only on the IBM-1 translation model; the second one combines the word alignment probability in IBM-1 and the phrases extracted from the HMM based word alignment model [10]; the third one is based on IBM-4 model in GIZA++ toolkit; and the last one is based on the translation templates (we use them as the translation phrase pairs in the statistical machine translation system) we extracted.

In our experiments, the BLEU score is used to evaluate the translation results (N=4).

The results of our experiment are shown in Table 3:

Table 3. Experimental results

Experiment	BLEU	NIST
1	0.2352	5.1533
2	0.2601	5.6309
3	0.2695	6.2178
4	0.2835	7.3582

The experimental results show that our method got the highest performance of the 4 systems.

There are still some problems on the quality of the translation templates we got. Two kinds of errors appear in the clustering results which affect the final results of the translation templates extraction. First, some errors occur in the grammar induction step: because the induction algorithm does not adequately use the information contained in the corpus, unrelated entities are sometimes clustered into a single group. The second sort of errors occurs in the alignment step when idiomatic translations are compared.

5 Conclusion

In this paper, we present an approach to automatic acquisition of translation templates from unannotated bilingual parallel corpora. The method is statistical and data-driven, and requires no parser. A grammar induction algorithm extracts from the corpus semantic and phrase structure grammars for both source and target languages. Based on these grammars, the phrase structures are aligned using BTG. Finally, the aligned structures are treated as translation templates. The preliminary experiment results show that the approach is effective.

However, we still face many difficult tasks, including the improvement of grammar induction and alignment. In the future, we will introduce more information and some additional pre-processing to improve the quality and efficiency of our approach.

References

1. H. Kitano. 1993. A Comprehensive and Practical Model of Memory-based Machine Translation. In 13. IJCAI. Chambery, France.
2. Satoshi Sato. 1995. MBT2: a method for combining fragments of examples in example-based translation. Artificial Intelligence, 75: 31-50.
3. Ilyas Cicekli and Halil Altay Guvenir, 2001. Learning Translation Templates from Bilingual Translation Exmples. In Applied Intelligence, Vol. 15, No. 1 pp. 57-76.

4. H. Watanabe, S. Kurohashi, and E. Aramaki. 2000. Finding Structural Correspondences from Bilingual Parsed Corpus for Corpus-based Translation. In Proceedings of the 18th International Conference on Computational Linguistics, pp 906-912.
5. K. Imamura. 2001. Hierarchical Phrase Alignment Harmonized with Parsing. In Proceedings of the 6th Natural Language Processing Pacific Rim Symposium, pp 377-384.
6. Dekai Wu, 1997. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Cor-pora. In Computational Linguistics, vol.23, No.3, pp. 377-403.
7. Yajuan Lü, Ming Zhou, Sheng Li, Changning Huang and Tiejun Zhao. 2001. Automatic Translation Template Acquisition Based on Bilingual Structure Alignment. Computational Linguistics and Chinese Language Processing. Vol.6, No.1, February, pp. 83-108
8. Franz Josef Och, Christoph Tillmann, Hermann Ney. 1999. Improved Alignment Models for Statistical Machine Translation. Proceedings of the Joint Conference of Empirical Methods in Natural Language Processing and Very Large Corpora; University of Maryland, College Park, MD, June pp. 20-28
9. Ye-Yi Wang and Alex Waibel. 1998. Modeling with Structures in Statistical Machine Translation. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics, Montreal, Canada. pp. 1357—1363.
10. Y. C. Zuo, C. Q, Zong, 2005. The method of extracting phrases based on HMM. In Proceedings of the 8th Joint Symposium on Computational Linguistics, Nanjing, China, pp.281-287.

HKUST/MTS: A Very Large Scale Mandarin Telephone Speech Corpus*

Yi Liu[1], Pascale Fung[1], Yongsheng Yang[1], Christopher Cieri[2],
Shudong Huang[2], and David Graff[2]

[1] Human Language Technology Center, Department of Electronic and Computer Engineering,
University of Science and Technology, Hong Kong
{eeyliu, pascale}@ee.ust.hk, ysyang@cs.ust.hk
[2] Linguistic Data Consortium, University of Pennsylvania, U.S.A.
{shudong, ccieri, graff}@ldc.upenn.edu

Abstract. The paper describes the design, collection, transcription and analysis of 200 hours of HKUST Mandarin Telephone Speech Corpus (HKUST/MTS) from over 2100 Mandarin speakers in mainland China under the DARPA EARS framework. The corpus includes speech data, transcriptions and speaker demographic information. The speech data include 1206 ten-minute natural Mandarin conversations between either strangers or friends. Each conversation focuses on a single topic. All calls are recorded over public telephone networks. All calls are manually annotated with standard Chinese characters (GBK) as well as specific mark-ups for spontaneous speech. A file with speaker demographic information is also provided. The corpus is the largest and first of its kind for Mandarin conversational telephone speech, providing abundant and diversified samples for Mandarin speech recognition and other application-dependent tasks, such as topic detection, information retrieval, keyword spotting, speaker recognition, etc. In a 2004 evaluation test by NIST, the corpus is found to improve system performance quite significantly.

Keywords: Mandarin, telephone speech.

1 Introduction

Speech database is the fundamental and important resource for spoken language processing technologies. The rich variations in human speech can only be adequately analyzed and represented in properly recorded, annotated and processed speech data. Currently, most of the state-of-the-art automatic speech recognition (ASR) algorithms are based on statistical approaches, which requires a large amount of training data with different ages, accents, speaking styles, speaking modes, channels, etc. to cover the diversity of human speech and speech environments.

There have been a lot of work and efforts in developing English and other major western languages [1, 2, 3]. These databases greatly facilitated the development of speech processing technologies, and many of them were published and released as

[*] HKUST Mandarin Telephone Transcripts, Part 1, Linguistic Data Consortium (LDC) catalog number LDC2005T32 and isbn 1-58563-352-6 http://www.ldc.upenn.edu/

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 724 – 735, 2006.

standard development and evaluation resources. The Linguistic Data Consortium (LDC) in the United States, founded in 1992, is a major center for supporting and coordinating corpora development activities. It has released more than 140 corpora both in speech and text, in more than 20 languages to over 750 organizations worldwide [1]. In Europe, a number of multilingual spoken language corpora have been developed with joint efforts from member countries in the European Union (EU). For example, SpeechDat has been created to support voice-activated applications over telephone with 20 regional variants of 14 major European languages [2, 3]. In Asian, Japan has invested heavily in the development of different types of Japanese databases, including telephony speech, lecture speech, broadcast speech, etc. [4]. Through the use of these databases, a lot of ASR systems have been established for practical use in Japan.

Chinese is one of the major languages in the world, and Mandarin (also known as Putonghua) is the official spoken language in mainland China, Hong Kong, Macau, and Taiwan. Mandarin speech recognition has attracted great interest in recent years. In particular, telephone conversational speech recognition is the latest pursuit by the ASR community since this type of speech is commonly used in daily life. SWITCHBOARD is a typical telephony conversational speech corpus that is widely used for English spontaneous ASR tasks [5]. CALLHOME Mandarin Chinese Speech is a corpus consisting of 120 unscripted telephone conversations between native speakers of Mandarin Chinese [1]. All calls were originated in North America and were placed to overseas locations, and most participants called family members or close friends. Similarly, the CALLFRIEND corpus includes both mainland and Taiwan dialects, which consists of 60 unscripted telephone conversations, lasting between 5 and 30 minutes. [1]. Both CALLHOME and CALLFRIEND were provided by LDC and released in 1996 and 1997. MAT is one of the first conversational Mandarin telephony speech corpus collected in Taiwan [6].

Compared to the English SWITCHBOARD corpus, there has been a dearth of data for telephone speech processing for Mandarin Chinese. CALLHOME and CALLFRIEND contain limited amount of data and most conversational topics are focused on family and school life, due to the nature of the calls. Since all subjects in the MAT are from Taiwan, most of them have a strong regional Min accent. Hence, the MAT corpus is not applicable to ASR systems for the majority of Mandarin speakers who are from mainland China. Therefore, it is desirable to develop a Mandarin conversational telephony speech corpus collected in Mainland China. It has become desirable to provide a speech database of a large number of native Mandarin speakers from mainland China, with high variations in age, occupational background, and education level. More importantly, the speech must be naturally spoken telephone conversations on a large number of different topics.

2 Phonological and Phonetic Properties of Mandarin Chinese

Acoustically and phonetically, Mandarin is quite different from European languages. The main differences are: (1) Chinese is monosyllabic; (2) Chinese characters are ideographic, and words consist of one or several characters. The pronunciation is represented by the syllable; and (3) Different characters may share the same syllable, this is known as homophony.

The pronunciation of Mandarin is represented by syllables. The structure of a syllable in Chinese is relatively simple: it consists of an *initial* and a *final*, or only the final. For standard Mandarin, there are around 1100 tonal syllables and 415 basic toneless syllables, 21 initials and 38 finals. Initials are very short in duration compared to syllables. Their pronunciations are rather flexible in spontaneous speech. Empirical results have shown that most of the pronunciation variations are caused by the changes in initials. In addition, there is one-to-many mapping between syllable and characters. On average, each syllable translates to 17 commonly used characters.

Mandarin is a tonal language. There are five lexical tones (including neutral tone) in [7, 8]. Each syllable is associated with a specific tone. The syllable with the same initial and final combination but with different lexical tones corresponds to different characters and has different meanings. Tones are a critical part of Chinese pronunciation and serve to differentiate meanings from characters of the same syllable.

Table 1. The dialect distribution in China

官话 Mandarin	闽 Min	湘 Xiang	赣 Gan	粤 Yue	客家 Kejia	吴 Wu
北东 N.E.	闽南 Minnan	土话 Tuhua	南昌 Nanchang	广州 G.Zh.	梅县 Meixian	太湖 Taihu
冀鲁 Jilu	蒲仙 Puxian	新湘语 N. X	鹰潭 Yingtan	五邑 Wuyi		台州 Taizhou
胶辽 Jiaoliao	闽东 Mindong	老湘语 O. X	抚州 Fuzhou			婺州 Wuzhou
北京 Beijing	闽北 Minbei		宜春 Yichun			处衢 Chuqu
中原 Central	闽中 Minzhong		吉安 Ji’an			瓯江 Oujiang
兰银 Lanyin	琼文 Qiongwen					宣州 Xuanzhou
西南 S.W.	邵将 Shaojiang					
江淮 Jianghuai						
桂柳片 Gui-Liu						

Unlike other languages, accent is a severe problem even for native Mandarin speakers. In addition to the standard Chinese Mandarin (Putonghua) spoken by radio and TV announcers, there are seven major language regions in China, including Guanhua, Wu, Yue, Xiang, Kejia, Min and Gan [7]. These major languages can be further divided into more than 30 sub-categories of dialects as shown in Table 1. In addition to lexical, syntactic and colloquial differences, the phonetic pronunciations of the same Chinese characters are quite different between Putonghua and the other Chinese languages. 70% of Chinese speakers on Mainland China are native speakers of Guanhua, the language group most related to Putonghua.

3 Corpus Design and Implementation

The HKUST/MTS corpus includes three parts: speech data, transcription and speaker demographic information. The speech data are recorded phone calls over public telephone networks. The calls are ten-minute natural spontaneous conversations between two native Mandarin speakers, who for the most part did not know each other prior to the recording. The recorded speech data are manually annotated with standard Chinese character transcription as well as specific spontaneous speech mark-ups. An MySQL database is established to store speaker's demographic information such as age, gender, dialect and language group, education background, phone types, background noise, etc.

3.1 Speaker

There are 2412 subjects in the corpus, of which 1252 are male and 1154 are female speakers. All subjects are native and fluent Mandarin speakers with nil or very slight accent. The speakers' age ranges from 16 to 60. The age distribution is shown in Table 2. The education background of the speakers ranges from high school to doctorate level. The occupation of speakers are as varied as government officers, bankers, university students, IT engineers, blue-collar workers, business people, etc. In addition, the birthplaces of all the speakers cover 221 cities or towns in 32 provinces of China. Each speaker is only allowed for one conversation recording.

Table 2. Speaker age distribution of the corpus

Range of speaker ages	Numbers	Distributions
<20	137	5.7%
20 - 24	942	39.2%
25 - 29	674	28%
30 - 34	524	21.8%
35 - 39	87	3.6%
>40	42	1.7%

3.2 Topic

In order to create a flexible, natural and creative conversational exchange representing a wide range of domains relevant to the Chinese culture, the conversations in the corpus cover wide topics. Unlike the CALLHOME and CALLFRIEND conversations, which tended to be all about family and school, our conversational topics are not "enforced", i.e., the call is included as long as there is continuous conversation on any topics. By consulting the English FISHER topics provided by LDC and with consideration to local cultural practice in mainland China, we finally provided 40 topics for the subjects to choose from. An example of selected topic list is shown in Table 3. The distribution of conversations over topics is listed in Fig.1. In general, there is a strong preference of the subjects for topics related to daily life, over political topics.

Table 3. Examples of conversation topics

Topic example	Contents of topic
1	**Life Partners --** What do you think is the most important thing to look for in a life partner?
2	**Computer games --** Do you play computer games? Do you play these games on the internet? What is your favorite game?
3	**Hobbies --** What are your favorite hobbies? How much time do each of you spend pursuing your hobbies?
4	**Travel --** Do you travel frequently? What are some of the advantages and disadvantages of traveling?
…	**……**

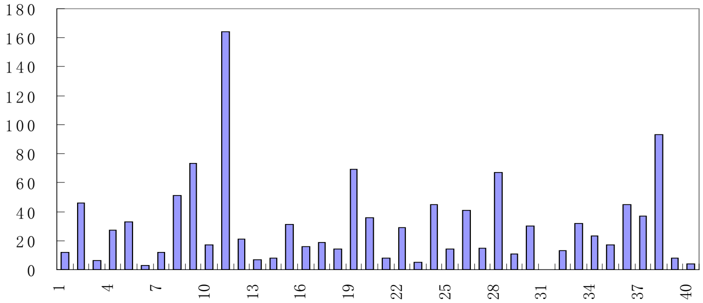


Fig. 1. The distribution of conversations over topics. The x-axis is the topic index and the y-axis is the number of conversations on that topic.

3.3 Channel

All data are recorded over public telephone networks, that is, it is recorded directly from telephone lines and not re-recorded or D-A-D converted data. Three telephone types are used in recording: fixed line phone, mobile phone and PHS phone. Fixed line phones include normal fixed line phone, IP phone and cordless phone. Mobile phones include GSM and CDMA formats, and all commonly used models such as Motorola, Nokia, Sony-Ericsson, Samsung, etc. In order to keep the diversity of telecom channel and line, the collected calls include local calls (two speakers are in the same city or town) and long-distance calls (two speakers are in different cities in mainland China). Each side of a call was recorded on a separate .wav file, sampled at 8 bits (a-law encoded), 8Khz. They were multiplexed later automatically in sphere format with a-law encoding preserved by using a tool developed in-house at HKUST. In the case where one side was shorter than the other, the shorter side was padded with silence.

3.4 Accent

Speakers in the corpus are all native and fluent speakers of Mandarin. However, in order to collect sufficient amount of speech samples from older speakers, senior citizen subjects with some accent are allowed to participate in the recording. Speakers with strong accent or whose utterances of Mandarin are too spontaneous to be intelligible are disqualified. Table 4 gives a detailed description of accent distribution of 2412 speakers in the corpus (based on speaker birthplace only). In Table 4, the second column illustrates general accent distribution of Mandarin speakers [7], while the third column gives the distribution in the HKUST/MTS corpus. Note that "Unknown" row means that there are 11 speakers in the corpus who did not provide the information or only provided vague information, such as "China".

Table 4. Accent distribution of the HKUS/TMTS corpus

Accent regions	Distribution in general public	Distribution in HKUSTMTS corpus
Guanhua	70%	77.1%
Wu	8.4%	8.8%
Cantonese	6%	8.4%
Xiang	5%	2.2%
Gan	2.4%	1%
Min	4.2%	1.7%
Kejia	4%	0.3%
Unknown	--	0.45%

3.5 Speaking Style and Speaking Rate

In order to collect natural spontaneous telephone conversations, subjects are asked to talk naturally, without trying to imitate broadcast news. The speaking volume is in general clear and steady. In general, speakers make about 70 to 90 utterances per ten-minute conversation under normal speaking rate, and most utterances are about 8 to 20 words. The length of one utterance should be less than ten seconds. The statistics of speaking rate in the HKUST/MTS corpus of 2412 speakers is illustrated in Table 5.

Table 5. Speaking rate information for HKUST/MTS corpus

Statistical criterion	Results
Average utterance length	4.6 s
Average character numbers per utterance	12.3
Average speaking speed	4.1 syllables/per second
Average utterance numbers per speaker in conversation	82

4 Corpus Collection

We use an operator-assisted recording approach for data collection. This is quite different from Fisher system and English EARS database collection system, which are automatic recording systems. Operator-assisted recording approach is the most reliable approach to get quality recordings and is particularly suitable for the Chinese collection due to the following reasons: (1) Many registered speakers will not respond to automatic calls. This has been the observation with past English database collection efforts; (2) Chinese are not used to respond to automatic messages; (3) High cost of automatic recording system; and (4) competitive labor costs in China. The procedure of recording is shown in Fig.2.

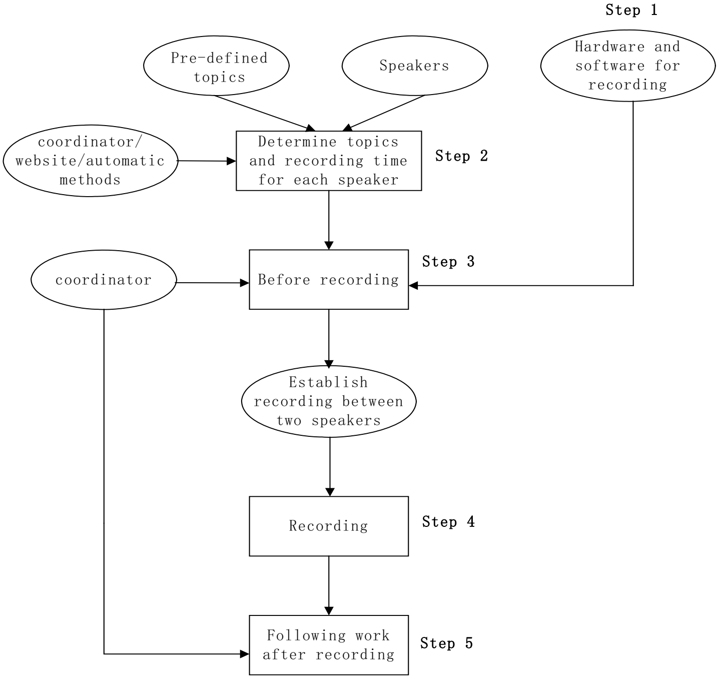

Step 1
Pre-defined topics
Speakers
Hardware and software for recording
coordinator/ website/automatic methods
Determine topics and recording time for each speaker
Step 2
coordinator
Before recording
Step 3
Establish recording between two speakers
Recording
Step 4
Following work after recording
Step 5

Fig. 2. The procedures of recording and recording management

4.1 Recording Conditions

In order to collect data with sufficient coverage, we set up 15 recording centers (including Beijing, Shanghai, Jilin, Xi'an, Jinan, Hefei, Nanjing, Shenzhen, Xiamen, Wuhan, Changsha, Chongqing, Xining, Nanchang and Fuyang) in mainland China covering all seven major dialectal regions of Mandarin speakers. Subjects were encouraged to make calls from home, from office or from any other relatively quiet environments. However, a small amount of ambient noise is acceptable. There is less than 10% of data of this type. In addition, only conversational topic and a brief instruction of how to use the recording system were provided to the subjects prior to the recording. Most importantly, speaking styles and vocabulary are not constrained.

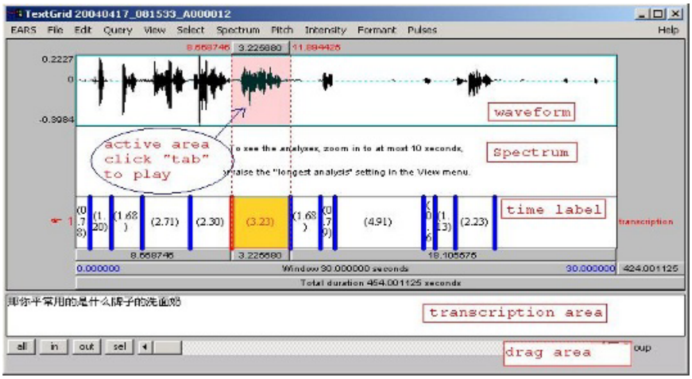
4.2 Recording Hardware and Software

We use Intel Dialogic E1 card for PRCD/600PCI-2E1 to connect China telecom E1 line with our recording servers. All data are recorded over public telephone networks. Speech from each speaker is recorded and saved to a wave file separately over a single channel. Our recording system supports 30 calls in parallel. The recruited subjects can either use fixed line phone or mobile phone/PHS phone to call the system for recording. In addition, the recording system supports local call, long distance call and international call.

The recording software include Dialogic card based software for conversation and speaker information recording; Interactive Voice Response (IVR) software for prompting questions and collecting speaker information; automatic speaker registration software for speaker registration and unique speaker ID allocation; communication software for two calls connection; separation software for two-channel recording separation and data saving; ftp server communication software for data transmission.

5 Transcription

The goal of transcription is to provide an accurate, verbatim transcript of the entire corpus, which is time-aligned with the audio file at the sentence level. Speech files are manually transcribed using standard simplified Chinese orthography in GBK code according to what the transcribers hear. Additional features of audio signal and speech are annotated with specific mark-ups for spontaneous speech. In general, we formulate transcription guidelines based on "LDC EARS RT-04 Transcription guidelines" [9]. The screenshot of a transcription tool developed in house for speech segmentation, labeling and transcription is shown in Fig.3. The transcriptions are saved in two formats: XML and TextGrid [10], which can be easily converted to any other format.



To see the analyses, zoom in to at most 10 seconds, or raise the "longest analysis" setting in the View menu.

Window 30.000000 seconds

Total duration 454.001125 seconds

那你平常用的是什么牌子的洗面奶

Fig. 3. An example interface for speech segmentation, transcription and time labeling

6 Post-processing and Analysis of Speech Data

Post-processing of recorded speech involves several rounds of fully manual and semi-automatic inspection and verification of the quality and the format of speech data as well as those of the transcriptions. In order to enable the HKUST/MTS corpus to be used for different speech applications, speech is segmented at natural boundaries wherever possible and each segment is no more than 10 seconds long. In addition, speaker demographic information is cross-validated based on the recorded waveform and information provided by the subjects. We analyzed the demographic distribution of recruited speakers every two months, and may reconsider the location of recruiting centers in order to achieve gender, age and accent balance.

6.1 Data Partition

LDC processed the HKUST/MTS corpus and separated it into three sets, namely the evaluation set, the development set and the training set [1]. The evaluation set contains 60 minutes of speech. This set includes 12 conversations by 24 speakers (equal number of male and female speakers). All subjects in the evaluation set are native Mandarin speakers in their twenties. In addition, all phones used are landlines, all conversations were recorded in mainland China, and all topics in the evaluation set were from current events and social issues [11]. The development set contains 120 minutes of speech. The selection strategy for this set is similar to that of evaluation set. The remaining data is used as training set for acoustic and language model estimation and generation.

6.2 Data Analysis

We give an initial analysis of the 200-hour collection to help readers better understand the corpus. All the conversations are segmented into utterances with length no more than 10 seconds long and transcribed at the Chinese character level with GBK encode format.

At the character level, the 200-hour corpus contains 248,910 utterances and 2,745,181 characters in total (filled pauses and spontaneous mark-ups are not counted). 3870 Chinese characters are used in the transcriptions. The auxiliary words "的" and "是" have the highest occurrence numbers at 91042 and 87734, respectively, which is in accordance with linguistic analysis [7]. At the syllable (Pinyin) level, since one character corresponds to one syllable, the corpus contains 2,745,181 syllables and covers all 408 toneless base syllables. At the initial and final unit level, all 27 standard Putonghua initials (including zero initials) and 38 finals are covered. A summary of the contents, syllable and initial final coverage of HKUST/MTS corpus is described in Table 6.

From the perspective of speech recognition, we are not only interested in how many units have non-zero occurrence numbers but also interested in how many of them have sufficient occurrences for robust acoustic model training. Ideally, we would like to have sufficient samples of all acoustic units.. In the corpus, it is found that 92%, 84% and 62.5% base syllables occur more than 100 times, 200 times and 1,000 times, respectively, and 18.25% base syllable have more than 10,000

Table 6. A summary of the contents, syllable and initial final coverage of HKUST/MTS corpus

Statistical criterion	Results
No. of utterances	248,910
No. of characters/syllables	2,745,181
No. of base syllable being covered	408
No. of standard initials being covers	27
No. of standard finals being covers	38

occurrences. Therefore, syllable-based acoustic modeling is also possible for many small and medium lexicon applications using HKUST/MTS as a training set.

Many state-of-the-art Chinese ASR systems use context-independent (CI) initial and final units instead of phoneme or phone as basic subword units for baseline acoustic model generation. Moreover, context-dependent (CD) acoustic modeling at sub-syllable level are widely used in ASR systems to achieve high recognition accuracy as well as good coverage of model complexity. Therefore, the occurrence number, the coverage and the distribution of different phonetic units of HKUST/MTS corpus need to be evaluated. Fig.4 gives the statistical distribution analysis of standard Chinese initials and finals. We can see that the distribution is in accordance with normal initial/final distribution, that is, the corpus is phonetically balanced. Due to the large amount of collected data, each context-independent or context-dependent subword units generated based on initial/final units has enough training samples. For example, even the least frequently used units “c” and “iong”, occurred 16147 and 4775 times respectively, which is enough for robust model generation.

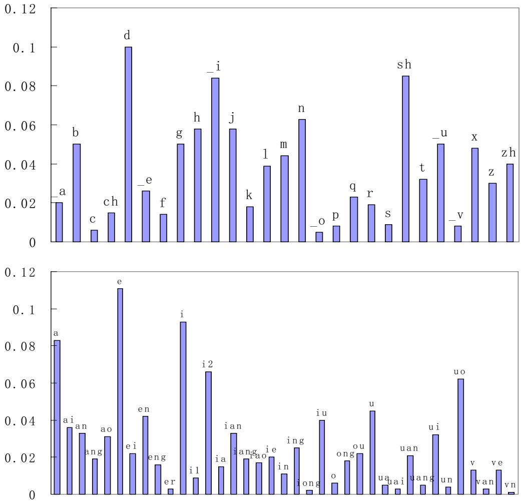


Fig. 4. Distributions of Chinese initials and finals in the HKUST/MTS corpus

Table 7 shows a detailed coverage of context-independent and context-dependent phonetic units. Since bi-phone level units as well as tri-phone level units are commonly used, both the intra-syllable contextual units and inter-syllable contextual units are considered. It is shown that all context-independent initial/finals are covered with sufficient samples, as well as for bi-phone level intra-syllable units. The HKUST/MTS corpus is an excellent resource for different types of ASR systems using different sub-syllable units, as well as with varied dictionary size for different applications. Furthermore, the high coverage of intra-syllable initial-finals and inter-syllable initial-finals means that the corpus is also suitable for robust triphone model generation and estimation.

Table 7. The phonetic coverage of HKUST/MTS corpus

No. of covered context-independent CI units		No. of covered CD intra-syllable units		No. of covered CD inter-syllable units	
Initials	21 (100%)	Initial-Final combinations	408 (99.7%)	Final-Initial combinations	788 (98.7%)
Zero initials	6 (100%)	Initial-Nucleus combinations	94 (100%)	Coda-Onset combinations	42 (100%)
Finals	38 (100%)			Tone-Tone combinations	20 (100%)

6.3 A First Evaluation

A first evaluation using the HKUST/MTS corpus was reported in the 2004 Fall Rich Transcription Speech-to-Text Evaluation of the National Institute of Standards and Technology (NIST) [11]. The evaluation set contains 60 minutes of speech data. This set includes 12 conversations with 24 speakers (equal number of male and female speakers), and the length of the conversation is five minutes. All speakers in the evaluation set are native Mandarin speakers in their twenties. The evaluation results reported are as shown in Fig.5. Three recognizers, BBN, Cambridge University and SRI, were used in evaluation. It is seen that the Character Error Rate (CER) is around 29%, which is much lower than the best performance of a similar evaluation in the previous year. This shows that the collected HKUST/MTS corpus used for training, development and testing has good quality and good phonetic coverage.

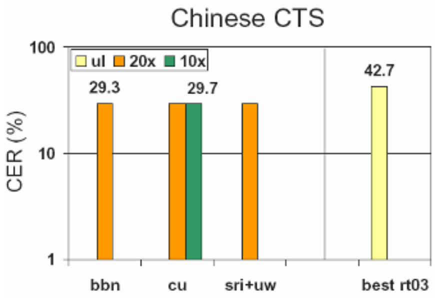


Fig. 5. The CER of BBN, Cambridge University and SRI recognizers evaluated using HKUST/MTS evaluation set

7 Conclusions

We described the design, collection, transcription and analysis of 200 hours of HKUST/MTS, a corpus of Mandarin Chinese conversational telephone speech from subjects in mainland China under the DARPA EARS framework. The corpus is the first of its kind that provides a large amount of naturally spoken conversations on a variety of pre-assigned topics. The corpus includes speech data, transcriptions and speaker demographic information. The corpus is an important resource for both application-specific and application-independent speech technologies, such as ASR, topic detection, pronunciation modeling, voice information retrieval and the analysis of conversational Mandarin. A large variety of acoustic speech samples from different telephone channels, including fixed line, IP phone, mobile phone and PHS phone, is included in the corpus. All speech data have been manually transcribed from the beginning to the end. Standard simplified Chinese characters, encoded in GBK were used. A tab-delimited tabular file with speaker demographic information was also provided. In addition, a software tool with relational database that has the functions of recording, multiplexing, transcribing, labeling, segmenting, checking and speaker information management has also been developed in-house at HKUST. A first evaluation of ASR tasks using this corpus has shown it to be very useful.

References

1. Linguistic Data Consortium (LDC), various corpus resources on http://www.ldc.upenn.edu.
2. European Language Resources Association (ELRA). http://www.elra.info/.
3. Hoge, H., et al.,: European speech databases for telephone applications. in Proceedings of the IEEE ICASSP, Vol.3, 1771-1774, 1997.
4. Ohtsuki, K., et al.,: Japanese large-vocabulary continuous speech recognition using a newspaper corpus and broadcast news. Speech Communication 28, 155-166, 1999.
5. Godfrey, J., et al.,: SWITCHBOARD: Telephone Speech Corpus for Research and Development. in Proceedings of the IEEE ICASSP, vol. 1, pp. 517-520, 1992.
6. Wang, H.C.: MAT- A project to collect Mandarin speech data through telephone networks in Taiwan. Computational Linguistics and Chinese Language Processing, Vol.2, No.1, pp73-90, 1997
7. Huang, J.H.:Chinese Dialects. Xiamen University Press, 1987 (Chinese version).
8. Lee, T., et al.,: Spoken language resources for Cantonese speech processing. Speech Communication 36, No.3-4, 327 - 342, March 2002.
9. LDC EARS Project RT-04 Transcription Guidelines: http://www.ldc.upenn.edu/Projects/Transcription/rt-04/RT-04-guidelines-V3.1.pdf
10. TextGrid as an objection of PRAAT: http://www.fon.hum.uva.nl/praat/manual/TextGrid.html
11. Le, A. et al.,: 2004 fall rich transcription speech-to-text evaluation. http://www.nist.gov/speech/tests/rt/.

The Paradigm for Creating Multi-lingual Text-To-Speech Voice Databases

Min Chu, Yong Zhao, Yining Chen, Lijuan Wang, and Frank Soong

Microsoft Research Asia, Beijing
{minchu, yzhao, ynchen, lijuanw, frankkps}@microsoft.com

Abstract. Voice database is one of the most important parts in TTS systems. However, creating a high quality new TTS voice is not an easy task even for a professional team. The whole process is rather complicated and contains plenty minutiae that should be handled carefully. In fact, in many stages, human interference such as manually checking or labeling is necessary. In multi-lingual situations, it is more challenge to find qualified people to do this kind of interference. That's why most state-of-the-art TTS systems can provide only a few voices. In this paper, we outline a uniform paradigm for creating multi-lingual TTS voice databases. It focuses on technologies that can either improve the scalability of data collection or reduce human interference such as manually checking or labeling. With this paradigm, we decrease the complexity and work load of the task.

Keywords: multi-lingual, text-to-speech, voice database.

1 Introduction

Most state-of-the-art text-to-speech (TTS) systems adopt concatenative speech synthesis approach, which perform unit selection in a large voice database, due to its capability in generating natural sounding speech. The naturalness of synthetic speech, to a great extent, depends on the size, the coverage and the quality of the voice database. Therefore, creating a high quality voice database is crucial for any unit-selection based TTS system. However, the whole process of database collection and annotation is rather complicated and contains plenty minutiae that should be handled carefully. In fact, in many stages, human interference such as manually checking or labeling is necessary. Creating a high quality new TTS voice is not an easy task even for a professional team. That's why most state-of-the-art TTS systems can provide only a few voices. In this paper, we outline a uniform paradigm for creating multi-lingual TTS voice databases with focuses on technologies that reduce the complexity and manual work load of the task.

(1) Be scalable. Though voice database is language dependent, we aim to have a platform scalable across different languages. To achieve this goal, it is very important to restrict language dependency within data and resources, such as text corpus, lexicon, and phone set. All algorithms used in the platform are designed language independent.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 736 – 747, 2006.

Beside the scalability in different languages, we also consider the scalability in different application scenarios. An important feature of concatenative TTS systems is that the synthetic speech inherits the voice characteristics and the speaking style of the voice talent who read the corpus. Therefore, just like different people have different taste for voice talents, TTS users often request for TTS voices that match their application scenarios. For example, industry users, who run TTS systems on servers and provide voice services to tens of thousands of people, usually prefer to have their specific voices attached to their brands or services. They would like to accept a large voice database but be critical on the quality of synthetic speech. Home users, on the other hand, often prefer to listen to known voices (from family members or close friends). In such cases, they normally can not afford to record a multi-hour speech database and their tolerance to distortions or errors in synthetic speech is higher than industry users. To serve for different requirements, scalability in creating voice database becomes important. In this paper, we introduce scalability in script generation and error detection.

(2) Minimize the labor intensive checking and labeling. Although, the script generation, phonetic transcription, unit segmentation and prosody annotation can be done fully automatically, the results are often not accurate enough. The errors in voice database will hurt the quality of synthetic speech when related units are used. Normally, manually checking or labeling is needed. In this paper, we discuss algorithms for unit segmentation and prosody annotation that can produce highly accurate results with limited amount of manually labeled training samples.

This paper is organized as follows. Section 2 describes the paradigm for creating multi-lingual TTS voice database. Section 3 to 6 introduces the key technologies that stress our research focuses. A summary is given in Section 7.

2 The Paradigm for Creating TTS Voice Databases

Though different TTS systems have different specifications for voice databases, they share common requirements for resources. As shown in Fig. 1, five types of data are normally needed, including the recording script, the recorded speech waveforms, the phonetic transcription, the segment boundaries and the prosody labels aligned to the speech waveforms. The corresponding processes, their key functions and challenges are described below.

2.1 Script Generation

The goal of script generation is to maximize the coverage of prosodic and phonetic variation of the base units in a limited amount of text script. Thus, at least three parameters, including the base unit set, the function for calculating coverage and the size of the script (or the total amount of speech planed to record), are to be decided according to the characteristics of the target language and the target scenario. Some experimental tips on how to decide the three parameters is introduced in the Section 3.

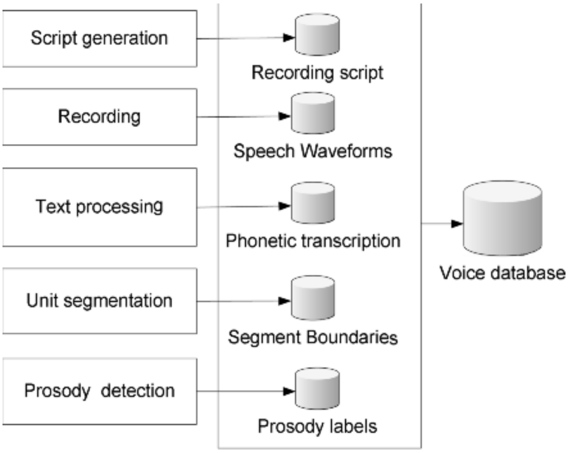


Fig. 1. Processes and resources needed for creating a TTS voice

After the three parameters are decided and all related language resources are available, a set of representative sentences can be selected from a large text corpus with a weighted greedy algorithm [1]

2.2 Speech Recording

The speech corpus for commercial usage is normally carried out by a professional team in a sound proof studio. The voice talent is carefully selected and well trained. With such constraints, the recorded speech, in general, have good quality. However, according to our experience, there are still about 1% words in the script been found not match with the speech. These mismatches are caused by reading errors, text-normalization errors and the idiosyncratic pronunciation of the speaker. For a personalized speech corpus that is recorded by a home user with a PC, the error rate will be much higher. These errors will hurt the speech quality when related units are used. Therefore, it is desired to have an automatic mismatch detection algorithm.

2.3 Text Processing

When generating the recording script and the corresponding phonetic transcription, many TTS front-end functions, such as the text normalization and the grapheme-to-phoneme conversion, are needed. These processes will often generate some errors and cause additional mismatch between the speech and its phonetic transcription. In Section 4, a generalized posterior probability based mismatch detection method [2] is presented.

2.4 Unit Segmentation

To make a speech corpus usable to a concatenative TTS, the phonetic transcriptions has to be aligned with the corresponding speech waveforms. HMM based forced alignment has been widely adopted for automatically boundary alignment [3]. Yet, despite its universal maximum likelihood and relatively consistent segmentation output, such a method can not guarantee the automatic boundaries are optimal for concatenation-based synthesis. Thus, post-refining is often performed to guide the boundaries moving toward the optimal locations for speech synthesis [4, 5]. Manually labeled boundary references are required to train the refining model. In Section 5, we propose to use context-dependent boundary models [6] to fine tune the segmental boundaries. Our goal is to improve the boundary accuracy with as fewer manual labels as possible.

2.5 Prosody Annotation

In order to achieve high quality synthetic speech, prosody annotation is often performed on the speech corpus, either manually or automatically. In most TTS systems, there is a prosody prediction module that predicts either categorical prosodic features, such as phrase boundary locations, boundary tone and pitch accent locations and types, or numerical features such as pitch, duration and intensity. Such prediction modules can be used to generate the prosody annotation for a speech corpus. However, the prediction from text quite often does not match the acoustic realization by the voice talent. In Section 6, we introduce a multi-classifier framework for automatic prosody annotation [7], in which the appearance of a prosodic event is jointly decided by an acoustic classifier, a linguistic classifier and a combined classifier.

Once all resources in Fig. 1 are available, the whole speech corpus or a selected part of it can be easily converted into a TTS voice automatically. Several key technologies that either improve the scalability or reduce the human inference are described in Section 3 to Section 6.

3 Choosing Proper Parameters in Script Generation

As mentioned in Section 2, three parameters have to decide before script generation. Some experimental tips are given below.

3.1 Size of the Script

Theoretically, the more speech recorded, the better the voice quality will be. However, in real-applications, we have only limited time and resources. In our previous study [8], the relationship between the size of speech database and the voice quality has been studied. As shown in Fig. 2, when about half of the corpus is pruned, the naturalness of synthesized speech is almost unchanged. If more than 70% of the corpus is pruned, the naturalness will drop rapidly. However, when 80% of the corpus is pruned, the MOS score is still above 3, which means acceptable. The speech corpus used in this study is in Mandarin and contains about 12000 utterances and 180,000 syllables. 20 percents of it equal to about 36000 syllables or 2400 sentences. This draws out the bottom-bound for the size of speech corpus.

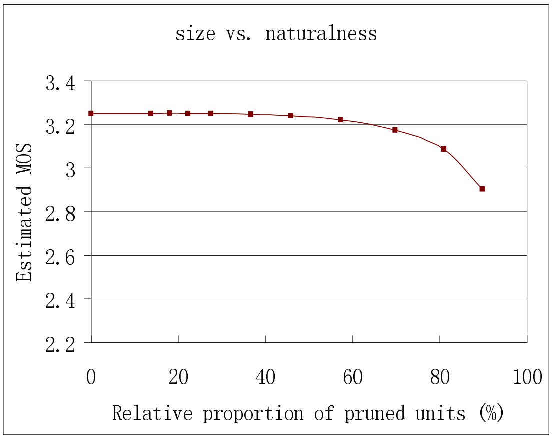


Fig. 2. The relationship between the size of speech corpus and the naturalness of synthesized speech. (the full size corpus contains about 180,000 Chinese syllables)

3.2 Base Unit Set

Base unit in a concatenative speech synthesizer is the smallest constituent in unit selection. There are many possible choices, such as phoneme, diphone, semi-syllable, syllable or even word. In order to get natural prosody and smooth concatenation, for each base unit, rich prosodic and phonetic variations are often expected. This is easy to achieve when smaller base units are used. However, there are several disadvantages for smaller base units. First, smaller units mean more units per utterance and more instances per unit and this implies a larger search space for unit selection and more search time. Second, smaller units cause more difficulties in precise unit segmentation and errors in unit boundaries will hurt the quality of the synthesized speech. It is found that longer base units are useful as long as enough instances are guaranteed to appear in the speech database [9].

For languages which have a relatively small syllable set, syllable is often used. For example, Mandarin Chinese has less than 2000 tonal syllables. Syllable can be used as the base unit if a more than 10 hour speech corpus is planed. However, when only a 1-2 hour speech corpus is planed, initial plus tonal final is an alternative choice. Since in zero-initial syllables and syllables with voiced initials, the segmentation between initial and final is very difficult, these syllables can be treated as whole units in a moderate size speech corpus, i.e. the base unit set can be a mixing of sub-syllabic units and syllabic units.

For most western languages, such as English, it is difficult to generate a closed list of syllables. Smaller base unit such as phoneme, diphone, halfphone are used. However, using phone-size unit will result a much larger search space for unit selection. Therefore, the return speed of the unit selection module in a phone-based English TTS system is dozens times slower than a syllable-based Mandarin TTS system (both systems share the same unit selection module). Therefore, it is highly recommended to expand the base unit set by adding some frequently used multi-phone unit. The criteria and the algorithm for identifying the multi-phone units are discussed in [9]. By keeping a good relationship between the number of multi-phone units added to the base unit set and the size of the speech corpus to be collected, the performance of the unit selection module can be kept in a good status in terms of both voice quality and return speed.

3.3 Function for Calculating the Script Coverage

The recording script should cover the most important phonetic or prosodic variations of each base unit. Some features like *position in phrase, position in word* and *position in syllable* (if sub-syllable units are used) are believed to cause variations in prosodic features of speech segments. Other features like *left phone type* and *right phone type* cause variations in segmental features. For tonal language like Chinese, the *left tone* and *right tone* should be considered. For languages that stress plays an important role, whether the unit is in a stressed syllable or not is an inducement for prosodic variations. When all these features are considered, they will result tens thousands possible contexts for each base unit. However, not all of them appear in real speech and the occurrence frequencies of context dependent units are non-uniformed. Therefore, we could generate a must-cover list that includes all context dependent units with occurrence frequencies higher than a threshold F, which is adjustable according to the target size of the script. Another constraint for the list is that each base unit has to appear for X times even if it has very low occurrence frequency. In order to cover all items in the must-cover list within the minimum size of script, the reciprocal of the occurrence frequency is often used as the importance index of each item. Then the importance of a sentence is measured by the sum of the importance indices of all new items it brings in. During sentence selection, the sentence with the highest importance is selected. After it is selected, all context dependent units in it should be removed from the must-have list. Then, the selection is repeated until the must-have list is empty or size of selected sentences reaches its up-bound.

In one experiment on Chinese script generation, a text corpus of five-year People's Daily, which contains about 97 million Chinese characters, is used as the raw corpus for statistic. Tonal syllable is used as the base unit. After all Chinese characters are converted into context dependent syllables, about 2.3 million distinct context dependent syllables are found. We found that the accumulated frequency of the top 44,000 items is larger than 50%. By setting 50% as the cutting threshold and constraining that at least 10 items per syllable, a must-cover list with 46,000 items is generated. After sentence selection, 12000 sentences are selected, which contains 177,000 Chinese characters and 119,000 distinct context dependent syllables. The additional 73,000 syllables raise the accumulated frequency to 64.0%. That is to say that we will have about 64% chances to find a syllable from the speech corpus with the required context during synthesis phase.

In another experiment, a small script is desired. We use initial and tonal final as the base unit. A 300-sentence script is generated, which contains about 6000 syllables.

4 Mismatch Detection

As mentioned in Section 2, both the recording process and text processing will generate mismatches between the speech and its phonetic transcription. Such mismatches will cause segmental errors in synthetic speech. They should be found in the data processing stage.

Generalized posterior probability (GPP), an integration of acoustic model score and language model score, is a probabilistic confidence measure for verifying the results of automatic speech recognition [10]. A typical usage of GPP is to verify the correctness of words [11], in which word GPP (GWPP) is estimated by exponentially reweighing the corresponding acoustic and language model likelihoods of all instances of a word in a word graph, as given in equation (1). GWPP has demonstrated robust performance on identifying the mismatches between script and speech that span to multiple syllables. However, it has rather high accepting ratio for wrong mono-syllabic words and words read slightly different from their canonical pronunciations. This is because of the usage of a word lexicon and the corresponding N-gram language model.

$$p([w;s,t] \mid x_1^T) = \sum_{\substack{M,[w;s,t]_1^M \\ \exists n,\ 1\le n\le M \\ w=w_n \\ |s_n-s|\le\Delta,|t_n-t|\le\Delta}} \frac{\prod_{m=1}^{M} p^{\alpha}\left(x_{s_m}^{t_m} \mid w_m\right)\cdot p^{\beta}\left(w_m \mid w_1^M\right)}{p\left(x_1^T\right)} \tag{1}$$

where [w; s, t] is the focused word w with its starting time s and ending time t, x_1^T is the sequence of acoustic observations, M is the number of words in the current string, α and β are the exponential weights for the acoustic and language models, respectively.

For a TTS database, besides the reading errors or orthographic error, local phonetic errors are desired to be identified. GPP for sub-word units is used for detecting such minor errors. Phoneme and syllable are two candidates. Since phonemes have very short duration, it is difficult to generate a reliable anchor for calculating GPP. Furthermore, the search space will be too large when all phoneme sequences are treated as legal. Thus, syllable is used in our work. First, a syllable lexicon and syllable N-gram are generated. Then, syllable-loop decoding is performed to generate syllable graphs. In order to get rich syllable hypotheses in the graph, only syllable unit-gram is used. An orthographic transcription is normally available for a TTS speech corpus so that the phonetic transcriptions can be derived from it. Next, the phonetic transcriptions are forced-aligned with the speech waveform. Finally, *generalized posterior syllable probability* (GSPP) for each syllable [2], defined as in equation (2), is calculated.

$$p([syl;s,t] \mid x_1^T) = \sum_{\substack{M,[syl;s,t]_1^M \\ \exists n,1\le n\le M \\ syl=syl_n \\ |s_n-s|\le\Delta,|t_n-t|\le\Delta}} \frac{\prod_{m=1}^{M} p^{\alpha}\left(x_{s_m}^{t_m} \mid syl_m\right)}{p\left(x_1^T\right)} \tag{2}$$

where [*syl*; *s*, *t*] is the focused syllable with its starting time *s* and ending time *t*, x_1^T is the sequence of acoustic observations, *M* is the number of syllables in the current string, α is the exponential weights for the acoustic models.

If the GSPP of a syllable is smaller than a threshold, *P*, the syllable is marked as not reliable. It will be excluded from the final database or be checked by a human annotator. By adjusting *P*, we are able to control the number of syllables to be withdrawn or checked.

5 Unit Segmentation

For a concatenative TTS system, the accuracy of unit boundaries is crucial to the voice quality of synthesized speech. The most commonly used method for boundary labeling is to perform HMM based forced-alignment. However, such alignments are obtained under the global maximum likelihood criteria, which do not guarantee the local optimum for concatenation. We propose a context dependent boundary model (CDBM) to refine the segmental boundary regarding to the boundary references provided by human.

Since the evolution of speech signal across a segmental boundary depends upon the phonemes before and after the boundary. A boundary point B, which is labeled by its left phoneme, X, and right phoneme, Y, and denoted as X-B+Y, is then denoted as a *Context Dependent Boundary* (CDB). As shown in Fig. 3, the characteristics of a CDB is represented by *(2N+1)* frames of acoustic features extracted from the frames spanning over a time interval across the boundary point and modeled by a *(2N+1)*-state HMM, where each state corresponds to one frame and the transition coefficient between neighboring states is always set to 1. Such a HMM is referred as the *Context Dependent Boundary Model* (CDBM). A certain mount of human labeled boundaries are required to train CDBMs. Ideally, one model per CDB. However, limited

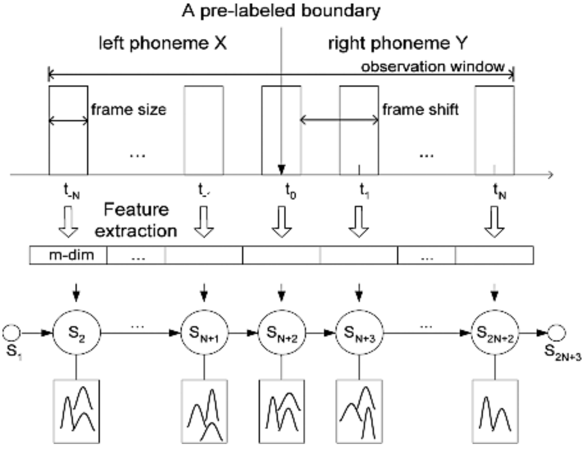


Fig. 3. Modeling of a context dependent boundary

segmented utterances in the training set cannot cover all possible boundary types. Therefore, CART is applied to cluster acoustically similar CDBMs to share data for robust estimation of parameters. As illustrated in Fig. 4, the clustering is performed on states. In total, *(2N+1)* CARTs are built, one for each state. A *Gaussian Mixture Model* (GMM) is then trained from all feature vectors clustered at the same leaf node. With such a hierarchical structure, the total number of CDBMs is adaptable.

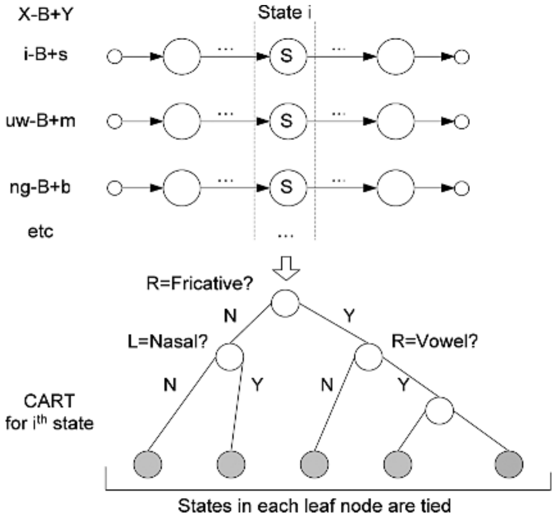


Fig. 4. Clustering CDBM states with CART

When such CDBMs are trained from 20,000 manually labeled boundaries, the boundary accuracy (if the distance from an auto-boundary to its manually labeled reference is smaller than 20ms, it is counted as a correct one) increases from 78.1% to 94.8% for Chinese and from 81.4% to 92% for English. If the training samples reduced to 5000 boundaries (approximately 350 Chinese utterances or 150 English utterances), the boundary accuracy is still above 90%. In the scenario to build personalized voice fonts, it is unrealistic to ask for manual labels, speaker independent CDBMs are trained. The boundary accuracy increases from 78.8% to 91.9% on the TIMIT testing set.

6 Prosody Annotation

Two types of prosodic events are normally labeled in a TTS speech corpus, the phrase boundary (w/o boundary type) and the pitch accent (w/o accent type). ToBI [12] is a widely adopted prosodic representation. It is first proposed for English and has been extended in many languages. However, annotating a speech corpus with ToBI is a very difficult task even for professionals. It will take even experienced labelers from 100 to 200 times real time [13]. The across personal agreement ratio for accent, edge tone and boundary indices are reported as 71%, 86%, and 74% respectively in [14].

However, the agreement ratio on the presence and absence of accent and edge tone are much high (92% and 93%, respectively). Therefore, a simple version prosody representation ToBI lite [15] is proposed recently. In our studies, we propose to annotate a set of prosodic event with complexity between ToBI and ToBI lite. It includes two-level boundary strength (correspond to the minor phrase and the major phrase boundaries), three boundary types (rising, falling and flat, correspond to the perceptual pitch movement before the boundary) and two-level accent (with/without accent). All these prosodic events have perceivable cues so that a well trained human annotator can achieve good self-consistency. In our experiment in English, the same annotator labeled the same sentences twice in a four-week time span. The agreement ratio on presence or absence of accent is 97.3%, on boundary strength plus boundary type is 97.1%. After the training section, labeling such prosodic events takes about 15 times real time. The cost for manual labeling is still high.

To reduce the human labeling efforts, we proposed a multiple classifier framework for prosody annotation [7]. As illustrated in Fig. 5, first, an acoustic classifier is used to detect the phrase boundary or the accented words from acoustic features and a linguistic classifier is used to predict phrase boundary or words to be accented from linguistic features. Then, the two results are compared. If they agree, the labels are kept. Otherwise, a third classifier is used to merge the scores from the two previous classifiers and some additional information such as word N-gram scores, segmental duration and pitch differences among succeeding segments. The third classifier gives the final labels on the disagreed part. With such a frame work, various acoustic features and linguistic features are combined together to make a more accurate prosodic labels.

In the experiment on accent labeling, the linguistic classifier is very simple, i.e. all content words are marked as with accent and all function words are without accent. The whole speech corpus was first labeled with the linguistic classifier and then used

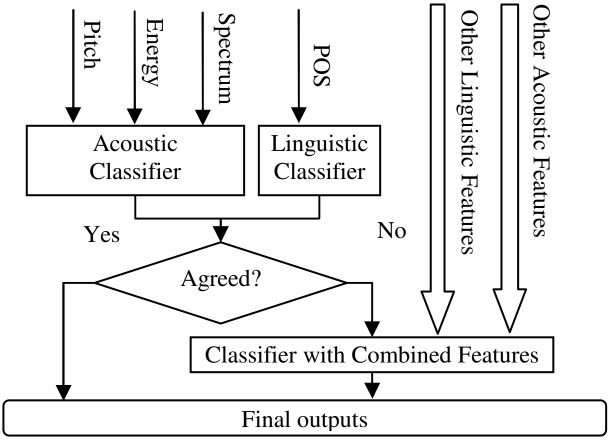


Fig. 5. The multi-classifier framework for prosody labeling

to train the acoustic classifier (HMM models were trained for accented or unaccented vowels). A small amount of manual labels are required to train the combing classifier (AdaBoost classifiers [16]). When 500 manually labeled sentences are used, we achieve 94.0% accuracy on the presence or absence of accent. Similar experiment is carried for boundary detection. The accuracy for presence or absence of phrase boundaries is 96.5%.

7 Summary

In this paper, the paradigm for creating multi-lingual text-to-speech voice databases is introduced with focuses on script generation, mismatch detection, unit segmentation and prosody annotation. With such a framework, we have created several large speech database (>10 hour speech), including two in Mandarin, one in English and one in Spanish. Besides, we have created 8 personalized speech databases, 4 in English and 4 in Chinese. These databases contain about 300 carefully selected sentences and read by our nice colleagues. With the technologies described in this paper, the work load and cost for developing a voice database is reduced. We hope this will inspire new request for using celebrity voice or personal voice, and voices in different speaking styles. As an initial attempt, we developed a demo which simulates a virtual chatting room on internet. Participants in the chatting room do not need to speak to their computer (though there are some voice-chatting rooms, many participants prefer to type in their words instead of speaking in to avoid disturbing others). Yet, the words they type in will be converted into speech with their voice fonts (or any pre-selected voice fonts). As the result, other participants in this room can still hear their words in their own voices, most likely through a headphone.

Acknowledgements. The authors would like to thank Eric Chang, Linda Chen, Frank Seide, Stephen Dahl and Sheng Zhao for their kindness to contribute their time and voices for the personalized voice databases.

References

1. Chu, M., Peng, H., Yang, H. Y. and Chang, E.: Selecting Non-uniform Units from a Very Large Corpus for Concatenative Speech Synthesizer, Proc. ICASSP-2001, Salt Lake City
2. Wang, L. J., Zhao, Y., Chu, M., Soong, F. K., Cao, Z. G.: Phonetic Transcription Verification with Generalized Posterior Probability, Proc. Interspeech-2005, Lisbon.
3. Toledano, D. T., Gómez, A. H.: Automatic Phonetic Segmentation, IEEE Trans. Speech and Audio Processing, Vol. 11 (2003) 617-625
4. Kominek, J., Bennet, C., Black, A. W.: Evaluating and Correcting Phoneme Segmentation for Unit Selection Synthesis, Proc. Eurospeech-2003, Geneva, 313-316
5. Adell J., and Bonafonte, A.: Towards Phone Segmentation for Concatenative Speech Synthesis, Proc. 5th ISCA Speech Synthesis Workshop, (2004)
6. Wang,L.J., Zhao, Y., Chu, M., Soong, F. K., Zhou, J. L., Cao, Z. G.: Context-Dependent Boundary Model for Refining Boundaries Segmentation of TTS Units, IEICE Transactions on Information and System, Vol. E89-D, N0. 3 (2006) 1082-1091

7. Chen, Y. N., Lai, M., Chu, M., Soong, F. K., Zhao, Y., Hu, F. Y.: Automatic Accent Annotation with Limited Manually Labeled Data, Proc. Speech Prosody 2006, Dresden.
8. Zhao, Y., Chu, M., Peng, H. Chang, E.: Custom-Tailoring TTS Voice Font – Keeping the Naturalness When Reducing Database Size, Proc. Eurospeech-2003, Geneva
9. Chen, Y. N., Zhao, Y., Chu, M.: Customizing Base Unit Set with Speech Database in TTS Systems, Proc. Interspeech-2005, Lisbon.
10. Soong, F.K., Lo, W.K., Nakamura, S.: Optimal Acoustic and Language Model Weights for Minimizing Word Verification Errors, Proc. Interspeech-2004*,* Jeju Island
11. Soong, F.K., Lo, W.K., and Nakamura, S.: Generalized Word Posterior Probability (GWPP) for Measuring Reliability of Recognized Words, Proc. SWIM-2004, Hawaii
12. Beckman, M. and Ayers Elam, G., Guidelines for ToBI Labeling, Version 3 (1997)
13. Syrdal, A.K., Hirschberg, J., McGory J., Bechman, M.: Automatic ToBI Prediction and Alignment to Speed Manual Labeling of Prosody, Speech Communication, Vol. 33, No. 1, (2001) 135-151
14. Syrdal A. K. and McGory, J.: Inter-Transcriber Reliability of ToBI Prosodic Labeling, Proc. ICSLP-2000, Beijing
15. Wightman, C. W., Syrdal, A. K., Stemmer, G., Conkie A., Beutnagel, M., Perceptually Based Automatic Prosody Labeling and Prosodically Enriched Unit Selection Improve Concatenative Text-to-Speech Synthesis, Proc. ICSLP-2000, Beijing.
16. Freund Y. and Schapire, R.E.: A Decision-Theoretic Generalization of Online Learning and an Application to Boosting, J. Comp. & Sys. Sci 55(1) (1997) 119-139

Multilingual Speech Corpora for TTS System Development

Hsi-Chun Hsiao, Hsiu-Min Yu[1], Yih-Ru Wang, and Sin-Horng Chen

[1] Department of Foreign Languages, Chung Hua University, Hsinchu
Department of Communication Engineering, Chiao Tung University, Hsinchu
schen@cc.nctu.edu.tw

Abstract. In this paper, four speech corpora collected in the Speech Lab of NCTU in recent years are discussed. They include a Mandarin tree-bank speech corpus, a Min-Nan speech corpus, a Hakka speech corpus, and a Chinese-English mixed speech corpus. Currently, they are used separately to develop a corpus-based Mandarin TTS system, a Min-Nan TTS system, a Hakka TTS system, and a Chinese-English bilingual TTS system. These systems will be integrated in the future to construct a multilingual TTS system covering the four primary languages used in Taiwan.

1 Introduction

The issue of collecting multilingual database has become popular in both automatic speech recognition (ASR) and text-to-speech (TTS) [4]. In TTS, multilingual corpora can be used to develop multilingual or polyglot TTS systems. They can also be used to analyze prosody behavior [2] in each individual language as well as in mixed languages, such as Chinese-English, for generating proper prosodic information to improve the naturalness of TTS systems. In ASR, multilingual corpora can be used to develop multilingual speech recognizers for cross language conversation or information retrieval applications.

In this paper, four speech corpora collected in the Speech Lab of NCTU in the past few years are introduced. They include a Mandarin tree-bank corpus, a Min-Nan corpus, a Hakka corpus, and a Chinese-English bilingual corpus. The main purpose of collecting these four corpora is to develop an integrated multilingual TTS system covering the main four languages used in Taiwan. In Taiwan, the official language is Mandarin Chinese with text written in Chinese character. But, there exist two popular dialects of Mandarin. One is Min-Nan which is the mother language of about 60% population. Another is Hakka which is the mother language of 11.5% population. These two dialects are widely used in the daily life of many people. Besides, the mixed speech of English and Mandarin is also used by many well-educated people. In this paper, we describe these four corpora in detail and discuss some of their uses in TTS.

The paper is organized as follows. Section 2 describes these four speech corpora. Section 3 presents some of their uses. Some conclusions are given in the last section.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 748 – 759, 2006.

2 The Four Speech Corpora

In this section, we describe the design, collection, processing and analysis of these four speech corpora in detail.

2.1 The Mandarin Tree-Bank Corpus

The Mandarin tree-bank corpus is composed of paragraphic read utterances of a single female professional announcer. It includes two sub-corpora. One contains utterances read in normal speed. It has 1422 sentences with 122,580 syllables or 70,910 words. The types of characters are 3133. Another contains speech of three repetitions of reading the same texts in three different speeds. It has 380x3 utterances (or 52,192x3 syllables) with three different speaking rates of 2.8, 3.5, 4.3 syllables per second.

Utterances of the corpus are all collected in the office environment and recorded in the form of 16 kHz sampling rate and 16-bit PCM format. The total memory size of the first sub-corpus is 1.01GB. All speech signals are automatically segmented into syllable sequence by the HMM method using the HTK and pitch-detected by using the ESPS. A part of the first sub-corpus, which contains 52,192 syllables, is further manually processed to correct the segmentation and pitch errors.

The text of each utterance in the corpus is a short paragraph composed of several sentences selected from the Sinica Tree-Bank Corpus [1]. Each sentence is associated with a syntactic tree parsed manually. Fig. 1 displays a typical example. With the syntactic information, we can explore the syntax-prosody relationship more deeply.

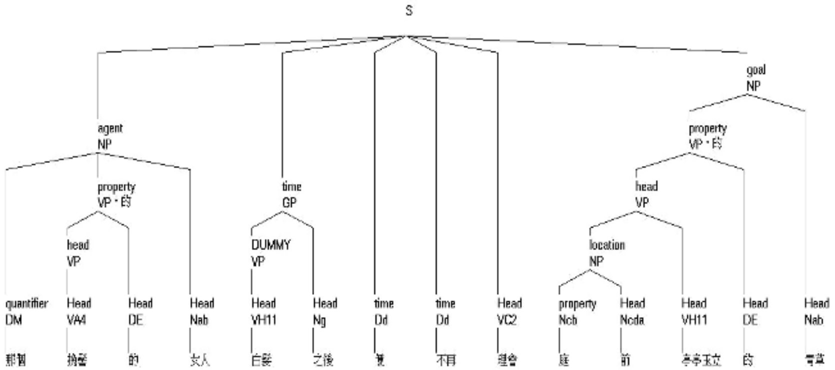


Fig. 1. The syntactic tree of an example sentence

2.2 The Min-Nan Speech Corpus

The Min-Nan corpus is designed for the development of a data-driven prosody generator for Min-Nan TTS. It consists of read speech of a single female speaker. It has 255 utterances including 130 sentential utterances with length in the range of 5-30 syllables and 125 paragraphic utterances with length in the range of 85-320 syllables.

The total syllable count is 23,633. The corpus is collected in the office environment and recorded in a 20 kHz rate. All speech signals are segmented into syllable sequences manually. Pitch is first detected by ESPS and then manually error-corrected.

All texts of the utterances in the corpus are represented in the Han-Lo (Chinese character-Roman alphabet) format [5]. They are processed manually. First, all Roman alphabet strings are converted into syllables. They are then segmented into word sequences and labeled with part-of-speeches (POSs). Besides, both the lexical tone and the sandhi tone (i.e., contextual tone change) of each syllable are determined manually. The main reasons of performing all these text processings manually are: (1) The standard written form of Min-Nan language does not exist. So the automatic tagging of text to obtain the word sequence is not easy. (2) Min-Nan speech has a complicated system of tone sandhi rules. The automatic labeling of text to determine the sandhi tone of each syllable is also not easy.

2.3 The Hakka Speech Corpus

The Hakka speech corpus is designed for the development of a data-driven prosody generator for Hakka TTS. It is collected in the office environment and articulated in Siixian accent (四縣腔) by a female informant, a retired elementary school teacher in her late fifty, and now a radio program hostess introducing Hakka dialect and culture. It consists of 59 read articles with the articles divided into 304 paragraphs to be fluently read by our informant. All speech signals are segmented into syllable by first using the HMM method and then manually verified. Pitch is first detected by ESPS and then manually error-corrected. For all the running speech, two things have been done: first, all the syllables in the speech, 42,011 in total, were transcribed in a modified version of Taiwan Tongyong Romanization, and the texts of the articles were parsed and tagged with parts of speech based on a simplified tagging set developed by Academia Sinica.

2.4 The Chinese-English Mixed Speech Corpus

The Chinese-English mixed speech corpus is designed for the prosody generation of Chinese-English bilingual TTS. The texts considered are all Chinese sentences embedded with English words. It consists of two sub-corpora. One is designed for the case of spelling English words such as “IBM” and “NBA”, while another is for the case of reading English words such as “Windows” and “Seven-Eleven”. The first sub-corpus consists of 539 utterances. The total syllable count is 13,540 including 1,872 English alphabets and 11,668 Chinese characters. Another sub-corpus consists of 423 utterances. It has in total 8,302 Mandarin syllables and 682 English words. The corpus is generated by a female speaker and recorded in the office environment. Utterances are all spoken naturally at a speed of 3.5 syllables/second. All speech signals are digitally recorded at a 20-kHz sampling rate. They were manually segmented into syllable sequences. Texts of the corpus are also manually processed to segment each sentence into word/POS/syllable sequences.

3 Some Uses of the Four Corpora

In this section, we present some uses of these four speech corpora.

3.1 Development of a Corpus-Based Mandarin TTS System

A corpus-based Mandarin TTS system is developed using the first sub-corpus of the Mandarin tree-bank corpus. Fig. 2 shows a block diagram of the system. The system is composed of four main modules including Text Analyzer, Prosody Generator, Unit Selection, and Waveform Synthesizer. Input Chinese text in the form of character sequence encoded in Big 5 format is first tagged by Text Analyzer to obtain the word, POS, character and syllable sequences. Unit Selection then uses the word sequence to search the speech segments of all partially-matching word strings from the texts of the corpus. These speech segments are taken as candidates of synthesis units to form a lattice. Meanwhile, prosodic information is generated by Prosody Generator using some linguistic features extracted from the word and POS sequences [9]. Then, the best speech-segment sequence to be concatenated to form the output synthesized speech is then found from the speech-segment lattice by the Viterbi algorithm. Fig. 3 shows a typical word-sequence lattice.

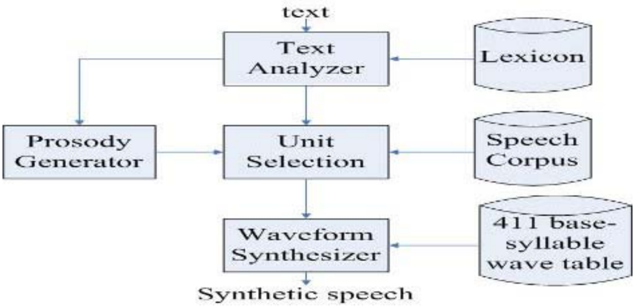


Fig. 2. The structure of the corpus-based Mandarin TTS system

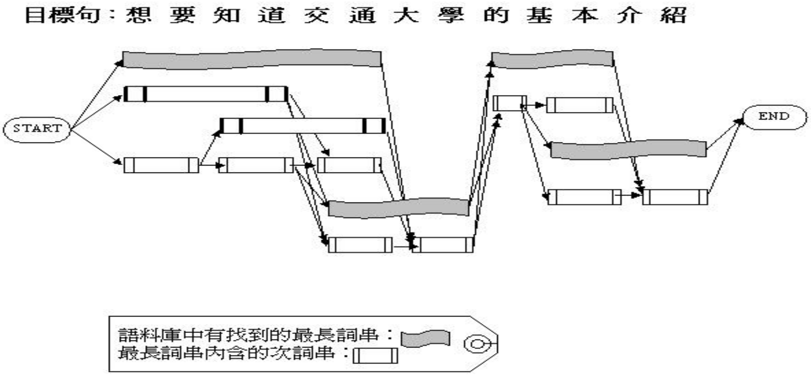

目標句：想 要 知 道 交 通 大 學 的 基 本 介 紹
START
END
語料庫中有找到的最長詞串：
最長詞串內含的次詞串：

Fig. 3. An example of candidate word-string lattice

The search of the optimal output speech-segment sequence is based on a cost function which considers both the intra-segment prosody matching cost and the inter-segment concatenation cost. The former measures the difference between the prosody of a candidate speech segment in the speech corpus and the target prosody generated by Prosody Generator. The latter measures the continuity of concatenating two candidate speech segments.

The function of Waveform Synthesizer is to generate the output synthesized speech by simply concatenating the optimal speech segments. For empty locations where no proper speech segments can be found in the speech corpus, waveform templates of base-syllable stored in an acoustic inventory are used to fill them. The PSOLA algorithm is employed to modify the prosody of those filling waveforms. Besides, a pause duration generated by Prosody Generator is inserted between two consecutive speech segments.

To improve the quality of the output speech, some special processings of the corpus are performed. Firstly, all long determiner-measure (DM) compound words are further segmented into small word segments. Secondly, prefix and postfix characters, surnames, and frequent monosyllabic words are collected and stored in a special acoustic inventory as synthesis units.

3.2 Development of a Min-Nan TTS System [7,12]

Min-Nan is a spoken dialect widely used in the south-eastern China and Taiwan. Just like Mandarin, Min-Nan speech is also a syllabic and tonal language [5]. There exist more than 2000 tonal syllables which are all legal combinations of 877 base-syllables and 8 tones including the degenerated Tone 6 which is not used by modern Taiwanese. These 877 base-syllables have almost the same initial-final structure like Mandarin base-syllables except that some base-syllables have finals with "stop" endings. Those special "stop"-ending base-syllables can only be associated with light tones (i.e., Tone 4 and Tone 8), referred to as entering tones. There are in total 18 initials and 82 finals.

Although Min-Nan speech has similar linguistic characteristics like Mandarin speech, it is a colloquial language and does not have a standard written form. There exist two popular written forms in Taiwan. One is the Romanization form which uses Roman alphabets to spell each base-syllable and uses a number to specify its tone. The other is a hybrid one in which most syllables are represented by Chinese characters with only a small set of special syllables being represented in Romanization form. Text written in this representation is easier to understand so that it is widely used in writing books and text documents. Unfortunately, the system to represent words in Chinese characters is still not standardized nowadays in Taiwan. Except some popular words, people always choose, according to their own preference, a string of Chinese characters with similar pronunciations in Mandarin to represent a Min-Nan word. This makes the text analysis of Min-Nan language very difficult because of the lack of a standard lexicon.

Another problem encountered in the text analysis of Min-Nan TTS is the determination of tone. Although there are only 7 lexical tones, the tone pattern of a

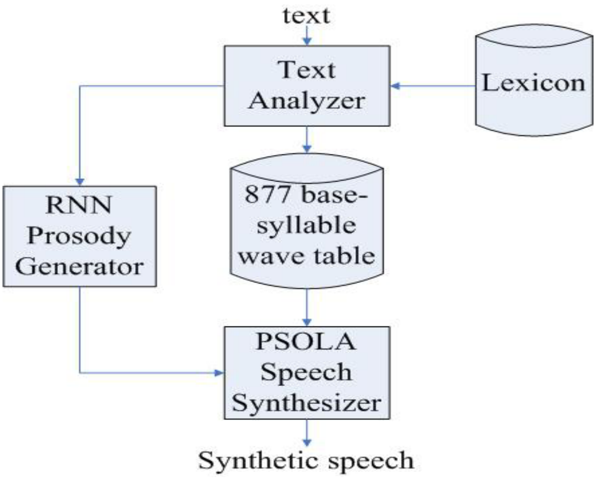


Fig. 4. A functional diagram of the Min-Nan TTS system (Source: Kuo [7])

syllable may change seriously in continuous speech. A previous study showed that Min-Nan speech possesses a set of tone sandhi rules [5]. Generally speaking, all syllables except the last one of a word chunk have to change their tones according a set of rules. So, the problem changes to a word chunking problem of determining sandhi word groups. A preliminary study of automatically labeling the tones of syllables was conducted by using the Min-Nan speech corpus [6]. It employed an F0 model to characterize the syllable pitch contour patterns and eliminate the interference from the high-level intonation. An average accuracy of 61.9% was achieved. With the automatic sandhi tone labeling, a further study to predict the boundaries of sandhi word groups from the input text will be done in the future.

A Min-Nan TTS system is developed by using the Min-Nan speech corpus [7,12]. Fig. 4 shows a block diagram of the system. It consists of four main functional blocks: Text Analyzer, RNN-based Prosody Generator, Acoustic Inventory, and PSOLA Speech Synthesizer. Input text is first tokenized into word/syllable sequence by Text Analyzer. The waveform sequence corresponding to the syllable sequence is then formed by table looking up from Acoustic Inventory which stores 877 waveform templates of base-syllable. Meanwhile, some linguistic features are extracted from the word/syllable sequence and used in RNN-based Prosody Generator to generate all required prosodic parameters. Lastly, a prosody modification of the waveform sequence to generate the output synthetic speech was performed by PSOLA Speech Synthesizer using these prosodic parameters.

To improve the quality of the output speech, some special processings of the corpus are performed. Firstly, a “Chinese-to-Min-Nan” lexicon is added to solve the out-of-vocabulary problem encountered in the text analysis using only a Min-Nan lexicon. This also makes the system possess the capability of processing input Chinese text. Secondly, a set of tone sandhi rules is explicitly applied in text analysis to change the lexical tones of all syllables into the ones for pronunciation. This makes

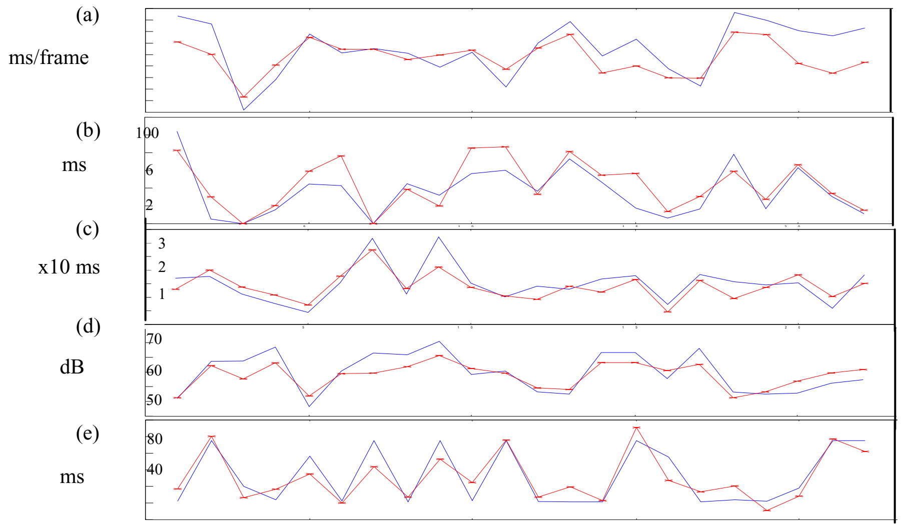


Fig. 5. A typical example of synthesized prosodic parameters: (a) pitch mean, (b) initial duration, (c) final duration, and (d) log-energy level of syllable, and (e) inter-syllable pause duration. The text is “生活應該是鮮豔、開朗、充實，自信滿滿the享受人生才著。”. (Source: Kuo [7])

the process of learning rules for generating proper prosodic information more easy. Thirdly, a further processing of the training speech database is done manually to label all major/minor breaks occurred at inter-syllable locations without punctuation marks (PMs) and to locate some special syllables pronounced short and lightly. Lastly, all 5- and 6-syllable words are classified into {2-3, 3-2} and {2-2-2, 3-3} pronunciation patterns. The new information obtained by these processings is used to help the prosody generation.

The whole system was implemented in software on a PC. Fig. 5 shows a typical example of the synthesized prosodic parameters. It can be seen from the figure that the synthesized prosodic parameters of most syllables matched well with their original counterparts.

3.3 Development of a Hakka TTS System [13]

Hakka is one of the major seven dialect families of Chinese spoken by native speakers in the provinces of southern China, Hong Kong, South-East Asia, and Taiwan. In Taiwan, it was estimated that Taiwan-Hakka is used approximately by two million native speakers, made up about 11.5% of Taiwanese inhabitants, hence a minor dialect used in Taiwan. Like many other minor languages/dialects in the world, Taiwan-Hakka is under the threat of disappearing from the pressure of dominant languages and gradual ethnic merging, which inevitably leads to a steep drop in the population of Hakka native speakers in the coming generations if no preventive measures are taken to avoid the disappearing of it or no adequate policy is made to encourage using, learning, and studying it. At this critical moment, it is the good timing to develop a Taiwan-Hakka text-to-speech system (THTTS) to facilitate the preservation and propagation of Hakka, and to expand the horizon of Hakka study and text-to-speech techniques as well.

To develop the THTTS along with its future practical application, we decided Siixian subdialect (四縣客語) as our model language, considering its largest population both in Taiwan and China and its accent viewed agreeably among Hakka folks to be the standard (but not necessarily the privileged) variety.

By using this corpus, we have constructed our first Hakka TTS system with RNN-based prosody generator and PSOLA-based speech synthesizer. The system is designed based on the same principle of developing the Min-Nan TTS system discussed in Section 3.2. Experimental results confirmed that the system performed well. An informal listening test shows that the synthetic speech sounds natural for well-tokenized texts, and fair for automatic tokenized texts. Further studies to improve the naturalness of the synthetic speech by incorporating a more sophisticated text analysis scheme and by adding some tone sandhi rules are worthwhile doing in the future.

3.4 Prosody Study on Chinese-English Mixed Corpus [8]

For the Chinese society, Chinese-English mixed texts or speech are very popular in Taiwan especially for the information processing domain. We give two examples:

> 我要進 IBM 公司。(I want to join the IBM corporation.)
> 送個 email 給我。(Give me an email.)

Besides, it becomes popular that young generations in Taiwan use short English alphabet strings to replace Chinese words as well as to represent some concepts for daily speech communication and for interactive communication through Internet. We list some of them in the following: BPP（白拋拋, white), SDD (水噹噹，very pretty) CBA (酷斃了, very cool), CKK (死翹翹, dead), LKK (老叩叩, very old), LM (辣妹, spicy girl), OBS（歐巴桑, mistress), OGS (歐吉桑, mister), PMP (拍馬屁, flatter), SYY（爽歪歪, very happy), etc.

In all applications of using Chinese-English mixed texts, Chinese is always the primary language. So the developments of Mandarin-English polyglot TTS systems are very urgent for Chinese societies. Now, we present an approach to expand an existing Mandarin TTS system [9] to a polyglot one which can properly spell English words letter-by-letter. The study focuses on the problem of generating proper prosodic information for English words in order to make their pronunciations match with the background Mandarin speech.

3.4.1 Analysis of Grammatical Constraints on the Chinese-English Mixed Speech Corpus [14]

It has been clearly proposed that the prosodic behavior of Chinese, for example, tone sandhi, is constrained by grammatical properties such as constituency and modification scope. According to Cutler [15], in speech comprehension, listeners do use explicit segmentation procedures. These procedures differ across languages and seem to exploit “language-specific rhythmic structure”. We may want to know what are the rhythmic segmentation procedures in Chinese-English mixed speech. By studying the way English words are incorporated into Mandarin sentences, we may get clues as to how “prosodic phrases” are formed in Mandarin-English mixed speech.

Chinese is a tonal language with clear tone patterns. Owing to this reason, when a speaker uses Chinese as his mother language, he will tend to pronounce English words with tonal concept. As a result, in Chinese-English mixed speech, some hypotheses are proposed:

- Speakers using Chinese as their mother language will tend to apply tonal concept to pronounce English words or characters, which makes English words sounds like consisting of several tonal syllables.
- Because English words are embedded in Chinese utterance, prosody information of English words will be guided by Chinese utterance in order to makes English words sounds not too strange and obvious.

Our hypotheses may be reasonable because of the following phenomena. When it comes to switching speaking language to English in a Chinese utterance, two language switching phenomena may happen. First is that when English words coarticulated with adjacent Mandarin syllables, it could be pronounced without any difficulty and sounds like constructed by tonal syllables. Another one is that if an English word can be constituent with adjacent Mandarin characters, it will be combined and becomes a prosody word, and will be sounded as natural as a Mandarin prosody word. These two phenomena shows that our hypotheses may be suitable for us to construct a Mandarin-English bilingual TTS system.

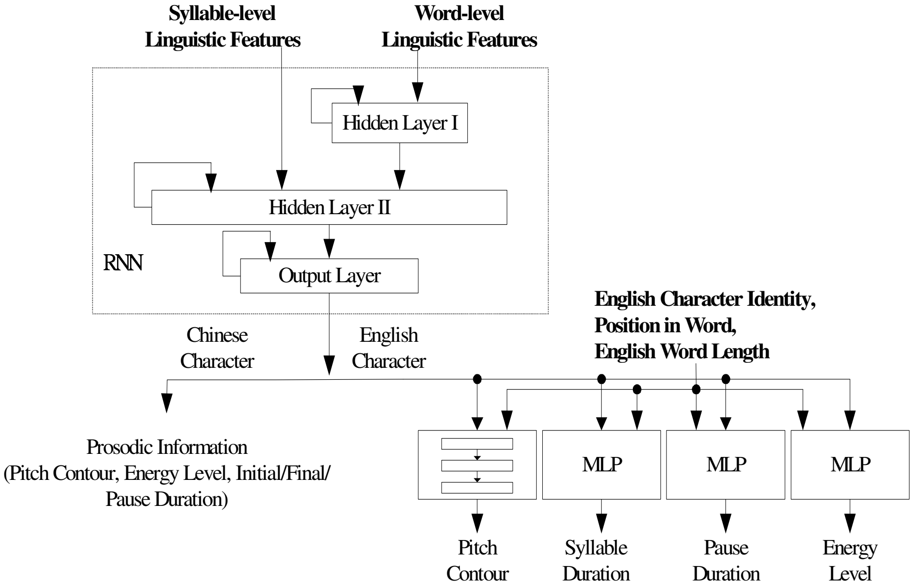


Fig. 6. A block diagram of the RNN-MLP scheme (Source: Kuo [8])

3.4.2 Prosody Generation for English-Chinese Bilingual TTS [8]

Fig. 6 is a block diagram of the proposed RNN-MLP scheme of generating prosodic information for spelling English words embedded in background Chinese texts. It adds four additional MLPs to follow the Mandarin RNN prosody generator developed previously. In operation, it first treats each English word as a Chinese word and uses the Mandarin RNN prosody generator to generate a set of initial prosodic parameters for each syllable of the English word. These initial prosodic parameters are expected to match globally well with those of the background Mandarin speech. It then divides these initial prosodic parameters into four subsets and employs four MLPs to refine them with the goal of compensating the distortions caused by the mismatch on the prosody pronunciations between the English word and the substituting pseudo Chinese word.

A typical example of the synthesized prosodic parameters for a Chinese-English mixed sentence is displayed in Fig. 7. It can be seen from the figure that all four initial prosodic features generated by the RNN for most English syllables matched well with the global trend of those for the background Mandarin speech. We also find from the figure that these initial prosodic features were greatly improved for most syllables by the four MLPs. This confirmed the efficiency of the RNN-MLP scheme. It can also be seen that the pre-English word pause duration at 是-USNS (Shi4-USNS) was lengthened while it was not for 則是-NB (Ze2 Shi4-NB).

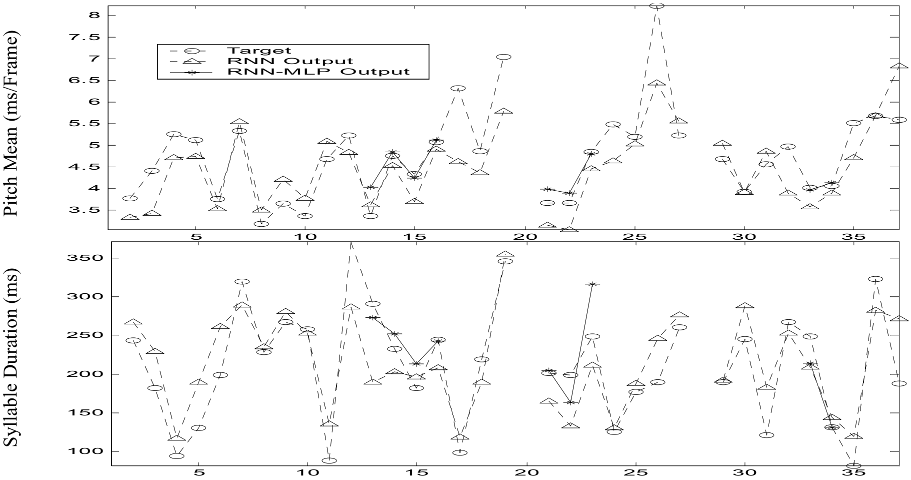


Fig. 7. A typical example of the generated prosodic parameter sequences of: (a) the pitch mean and (b) duration of syllable. The text is: "現在的年輕人，身上穿的是，USNS的衣服，IBS的牛仔褲，腳上則是NB的鞋子。". (Source: Kuo [8])

4 Conclusions

In this paper, four speech corpora collected in NCTU and their uses have been discussed. Using these four corpora, three individual TTS systems for Mandarin

Table 1. The RMSEs of the synthesized prosodic parameters

	Mandarin		English-Chinese Bilingual				Taiwanese		Hakka	
	Inside	Outside	word spelling		word reading		Inside	Outside	Inside	Outside
			Inside	Outside	inside	Outside				
Pause (ms)	23.7	54.5	25.9	30.8	25.7	26.0	27.62	29.06	56.8	65.4
Initial (ms)	17.2	18.5	13.4	14.5	21.6	23.8	12.00	12.19	20.7	25.6
Final (ms)	33.3	36.7	35.2	37.6	41.3	45.5	34.55	35.22	42.9	45.7
Pitch (ms/frame)	0.84	1.06	0.56	0.65	0.45	0.45	0.84	0.85	1.9	2.2
Energy (dB)	3.39	4.17	2.16	3.06	3.41	4.90	2.53	2.97	3.7	4.3

Chinese and its two dialects, Min-Nan and Hakka, and a Chinese-English bilingual TTS system were developed, and the RMSEs of synthesized prosody parameters are displayed in table 1. The research will be continued to integrate these TTS systems into one multilingual TTS system to cover the four main languages used in Taiwan.

Other uses of these four speech corpora on prosody modeling to exploit the relationship between the hierarchical prosody structure and the hierarchical linguistic structure of Mandarin Chinese is now under studied [3,10,11].

Acknowledgement

This work was supported in part by MOE under contract EX-94-E-FA06-4-4 and in part by Council for Hakka Affairs. The authors want to thank the MediaTek for supporting our research and the Academia Sinica, Taiwan for providing the Sinica Tree-Bank Corpus. The authors thank W. Z. Gong for supplying the lexicon and the text corpus.

References

1. Huang, Chu-Ren, Keh-Jiann Chen, Feng-Yi Chen, Keh-Jiann Chen, Zhao-Ming Gao and Kuang-Yu Chen. 2000, “Sinica Treebank: Design Criteria, Annotation Guidelines, and On-line Interface”, Proceedings of 2nd Chinese Language Processing Workshop 2000, Hong Kong, pp. 29-37.
2. Colin W. Wightman, Mari Ostendorf, “Automatic Labeling of Prosodic Patterns”, IEEE Transactions on Speech and Audio Processing, vol. 2, no. 4, October 1994, pp. 469 – 481.
3. Chiang, Chen-Yu, Wang, Yih-Ru and Chen, Sin-Horng (2005), "On the inter-syllable coarticulation effect of pitch modeling for Mandarin speech", INTERSPEECH-2005, pp. 3269-3272.
4. Fu-Chiang Chou, Chiu-Yu Tseng, Lin-Shan Lee, “A Set of Corpus-Based Text-to-Speech Synthesis Technologies for Mandarin Chinese”, IEEE Transactions on Speech and Audio Processing, Volume 10, Issue 7, Oct 2002, pp. 481- 494.

5. R. L. Cheng, Taiwanese pronunciation and Romanization – with rules and examples for teachers and students, Wang Wen Publishing Company, 1993.
6. Wei-Chih Kuo, Yih-Ru Wang, and Sin-Horng Chen, "A model-based tone labeling method for Min-Nan/Taiwanese speech", ICASSP 2004.
7. Wei-Chih Kuo, Xiang-Rui Zhong, Yih-Ru Wang and Sin-Horng Chen, "High-Performance Min-Nan/Taiwanese TTS System" , ICASSP2003.
8. Wei-Chih Kuo, Li-Feng Lin, Yih-Ru Wang, and Sin-Horng Chen, "An NN-based Approach to Prosodic Information Generation for Synthesizing English Words Embedded in Chinese Text" , in the proc. of Eurospeech2003.
9. S. H. Chen, S. H. Hwang, and Y. R. Wang, “An RNN-based Prosodic Information Synthesizer for Mandarin Text-to-Speech”, IEEE Trans. Speech and Audio Processing, Vol.6, No.3, pp.226-239, May 1998.
10. S.-H. Chen, W.-H. Lai, and Y.-R. Wang, “A New Duration Modeling Approach for Mandarin Speech,” IEEE Trans. On Speech and Audio Processing, vol. 11, no. 4, July 2003.
11. Sin-Horng Chen, Wen-Hsing Lai and Yih-Ru Wang, “A statistics-based pitch contour model for Mandarin speech,” J. Acoust. Soc. Am., 117 (2), pp.908-925, Feb. 2005.
12. S. H. Chen and C. C. Ho, “A Min-Nan Text-to-Speech System,” ISCSLP’2000, Beijing, China, Oct. 2000.
13. Hsiu-Min Yu, Hsin-Te Hwang, Dong-Yi Lin and Sin-Horng Chen, “A Hakka Text-to-Speech System,” submitted to ISCSLP2006.
14. S. H. Chen, et al, Technical Report of NCTU, MOE project EX-94-E-FA06-4-4.
15. Cutler, A., & Otake, T. (2002). Rhythmic categories in spoken-word recognition. Journal of Memory and Language, 46 (2), 296-322.

Construct Trilingual Parallel Corpus on Demand

Muyun Yang, Hongfei Jiang, Tiejun Zhao, and Sheng Li

MOE-MS Key Laboratory of Natural Language Processing and Speech
School of Computer Science and Technology
P.O. Box 321, Harbin Institute of Technology, Harbin, 150001
{ymy, hfjiang, lisheng, tjzhao}@mtlab.hit.edu.cn

Abstract. This paper describes the effort of constructing the Olympic Oriented Trilingual Corpus for the development of NLP applications for Beijing 2008. Designed to support the real NLP applications instead of pure research purpose, this corpus is challenged by multilingual, multi domain and multi system requirements in its construction. The key issue, however, lies in the determination of the proper corpus scale in relation to the time and cost allowed. To solve this problem, this paper proposes to observe the better system performance in the sub-domain than in the whole corpus as the signal of least corpus needed. The hypothesis is that the multi-domain corpus should be sufficient to reveal the domain features at least. So far a Chinese English Japanese tri-lingual corpus totaling 2.4 million words has been accomplished as the first stage result, in which information on domains, locations and topics of the language materials has been annotated in XML.

Keywords: trilingual corpus, corpus scale, multi-domain, machine translation.

1 Introduction

To provide necessary information for anybody at any time in any location is one of the most challenging tasks faced by Beijing 2008. The NLP systems are the most promising solution to this problem. Current NLP technology can be characterized by so-called corpus approach. That is to say, the NLP system demands a reasonable collection of language material as the training data. Therefore, the performance of the NLP system for Beijing 2008 relies heavily on the quality and scale of the training corpus available.

To acquire the proper corpus for the NLP system development is not a trivial task. Although language is reproduced and recorded in an extremely large amount everyday, it does not readily exist in proper quality and right form for a corpus purpose. In a sense, corpus collection technology itself remains an open issue in NLP research [1].

So far, the reported corpora are chiefly constructed for NLP researches instead of practical application system development. For example, most of the famous corpus projects like LOB, BNC, Brown Corpus and Penn Treebank are all designed for the research of a NLP subtask [1]. Another kind of corpus construction is closely related

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 760 – 767, 2006.

with public evaluation of certain kind of NLP technique. For instance, the accumulation of Chinese-English parallel corpus is devoted to evaluate MT (machine translation) performance in a large scale. These efforts in corpus collection make no attempt to answer how much data should be enough. Without the burden of ensuring a successful development of a NLP application, they either work to the time and money allowed, or continuously accumulate all materials available, scaling up into a huge enough corpus in the long run.

In contrast to these corpus constructions, the Olympic Oriented Parallel Corpus presented here is designed to support the NLP development from the very beginning. Challenges of the corpus come from 2 aspects: 1) In contrast to a given NLP sub-task, Beijing 2008 requires many kinds of NLP systems, any of which generally integrates several NLP sub-technologies; 2) Beijing 2008 also involves many domains of language. To enable the same good performance over all the domains, current NLP technologies, which are chiefly based on statistical model, require a sufficient training corpus on each domain. The key to these problems is how to provide suitable data in a proper scale demanded by the NLP system development.

In order to solve this problem, this paper describes a typical application based corpus scale measure. Instead of trying to figure out how much data is enough, the proposed measure tries to figure out the "minimal data needed" by the hypothesis that a multi-domain corpus should, at least, reveal the features of each domain. The rest of the paper is arranged as follows: section 2 briefly introduces the overall design of the corpus. Section 3 presents the corpus scale control strategy based on application performance observation in each domain vs. in the whole corpus. Section 4 gives the experimental results of our method and describes the present progress of the Olympic Oriented trilingual corpus. And, finally, section 5 concludes this paper.

2 Design and Collection of Olympic Oriented Parallel Corpus

Since the chief task of the NLP application for Beijing 2008 is to provide information access for the people coming from the world, cross language information service system becomes the chief development task. Therefore, the parallel corpus that consists of several languages as mutual translations is the target of the construction. In practice, multi-lingual corpus is a challenge since parallel corpus is extremely rare in existence. To facilitate the corpus construction, Chinese, English and Japanese are chosen as the first targets because they are most easily accessed in China. And the purpose is to collect the “Olympic Oriented Trilingual Corpus” to support related NLP application development.

It can be further noticed that the “Olympic” is actually a social event rather than a single language domain. Hence the parallel corpus should contain all domains of language involved in the event. After discussion with NLP system developers, altogether 5 most related domains are chosen to be included in the first stage of the corpus construction, i.e. traveling, food&drink, traffic, business and sports. Although both speech and text material are necessary, only text material is focused for the current construction to meet the urgent needs of cross-lingual service development.

As a compensation strategy, transcripts of dialogues are more emphasized than essay (written articles).

Ideally the corpus should be collected from real language usage. But direct record of such language material, such as the building of trilingual spontaneous speech corpus mentioned in [2], is extremely expensive and time consuming. As an alternative, we first try to get the parallel corpus from the Internet. Though parallel corpus mining from the Internet is reported useful [3,4], we found that it is not suitable for our purpose because:

1) The amount of the parallel corpus of the 5 domains mentioned above is limited;
2) Manual verification of the automatically collected texts is not an easy task;
3) Most of the parallel corpus from the Internet is selected form English textbook, with many typos in the texts;

As a result, we turn to traditional way of selecting language material from the publications. Manual input is carried out, and verification is conducted with corpus annotation. Figure 1 displays the main steps of the corpus construction.

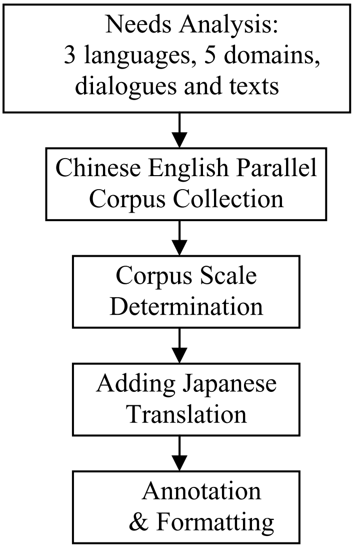


Fig. 1. Main Stages of the Trilingual Corpus Construction

The trilingual corpus is aligned at the sentence level, which is manually checked. As for its annotation, the following information is included:

1) Genres: dialogue, article or sample sentence;
2) Scenario : the place where the dialogue happens or that the article is describing;
3) Topic: the topic of the dialogue or the articles;

A summary of the scenario tags and topic tags is given in Table 1 of Section 4.

3 Application Performance Based Corpus Scale Determination

The key to the corpus collection is to determine how much data should be contained. Ziff law indicates that there are always half low frequency words in the corpus no matter how large the corpus is. So people now don't attempt to build a perfect large corpus any longer. But in practice, "how much is enough" can not be avoided in corpus collection.

A method to decide the corpus scale is to observe the improvements of the model performance with the data expansion. Usually the model performance will reach the ceiling after a first phase of increase brought by the data expansion. Although this information is usually provided for NLP model development, it might as well be treated as a signal of sufficiency of corpus data.

To adopt this method in the trilingual corpus collection aimed at a specific application development, there exist the following disadvantages:

1) The test suite should be chosen properly to measure the real performance of the system, which is not an easy task in itself.

2) Using this method implies that the target systems have already been established for testing purpose. However, we are now building a corpus for the subsequent system development.

3) Cross language services in Beijing2008 involves many NLP technologies. In practice we cannot test them one by one on the corpus.

4) The corpus covers 5 domains as mentioned above and each scale needs to be justified as large enough.

It’s lucky that the first stage of our trilingual corpus construction is to provide “reasonable large data”. For this purpose, this paper proposes a corpus scale measure based on the application performance over the multi-domain parallel corpus. The hypothesis is that, when dealing with multi-domain corpus, the least amount of the corpus scale should be such that each domain would not lost its own features against the whole. A way to measure this is to check the system performance in each domain against that in the whole corpus. In detail, the strategy contains the following steps:

1) Choose the system from the candidate applications;

2) Get the test sets and evaluate the system performance for each domain of the corpus;

3) Combine the test sets and evaluate the system performance in the whole corpus;

4) Compare the differences of the above tests in relation to the corpus expansion, and the point where domain performance surpasses the whole corpus performance is the least scale of the whole corpus.

Usually, if a NLP task is more difficult and the model does less generalization, the system would require more training data. As far as the Olympic Oriented Trilingual Corpus is concerned, machine translation (MT) is perhaps the most challenging task among other cross language NLP systems to be developed. And in the 3 main models of MT, i.e. rule based MT (RBMT), example based MT (EBMT) and statistical MT (SMT), EBMT adopts a "case-to-case" strategy and thus provides less data generalization compared with the other two. So the corpus scale can be safely measured by the EBMT performance.

4 Corpus Scale Determination by Chinese-English EBMT

The EBMT system chosen for the experiment in this paper is a word-alignment based Chinese-English EBMT[5]. It is a pure EBMT system without any hybrid module combination, which can be built fully automatically from the word aligned bilingual corpus. To automatically evaluate the EBMT system performance, one of the most popular MT evaluation standards 5-gram NIST score is adopted [6].

During the corpus collection, 1000 Chinese English sentence pairs are called as a section. The 5 domains are enlarged section by section simultaneously. Section 0 of each domain is reserved as the test sets of corresponding domain. The later sections are treated as the training data. When the corpus is enlarged section by section, the system performance on the 5 domain is observed by two translation conditions: 1) translated by the example base built from the section(s) from corresponding domain; 2) translated by the example base built from the section(s) of the 5 domains.

In the beginning of our experiment, the EBMT performs better in the whole corpus than in the specific domain on each test set. But the two results get close when more sections are added into the each domain. When the whole corpus are enlarged into 50,000 sentence pairs, i.e. 10 sections for each domain, the EBMT built from the specific domain slightly produces a better results than the whole corpus. To verify this result, we did ten-fold cross validation for each domain: selecting each section as the test set and observe the EBMT performance over the rest 9 sections. Fig2-6 shows the average results of the 5 domains, in which the horizontal axis represents the number of sections used to build the translation example base, and the vertical axis represents the quality of the translation result in NIST score.

As shown in Fig 2-6, the domain of food &drink, sports and traffic succeed in producing better translation result against the whole corpus. But the domain of traveling and business fail to produce better results. After further analysis, the reasons for such failure can be summarized as the following:

1) The domain of business is originally designed for transcriptions of business related dialogues. But, owing to the scarcity of such materials, business letters are allowed. In fact, the two kinds of material preserve rather different language usage. The dialogue usually produces simple short sentences, which the business letter often contains complex long sentences. The whole business domain can actually be treated as two sub-domains owing to different genres. With roughly 5000 sentence pairs divided for each, it is reasonable that the two sub-domains are in need of more data to reveal their features, let alone the business as a whole to bring its own feature against the whole corpus.

2) The case for the domain of traveling is a bit more subtle. In facts, tourism-related language materials are all thrown into this domain except those with clear mark for the other four domains. The language materials without clear scenarios are also grouped into here, such as greeting expressions that can be used in either restaurant, sports stadium or during a trip. In this case, the traveling domain would be like a salad of the other four domains, mostly represented by language materials on hotels, post offices, hospitals, etc.

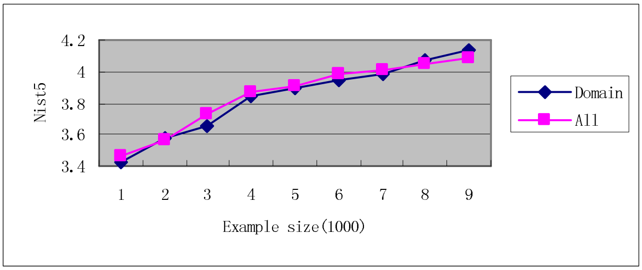


Fig. 2. EBMT Performance in Food&Drink Domain vs. in Whole Corpus

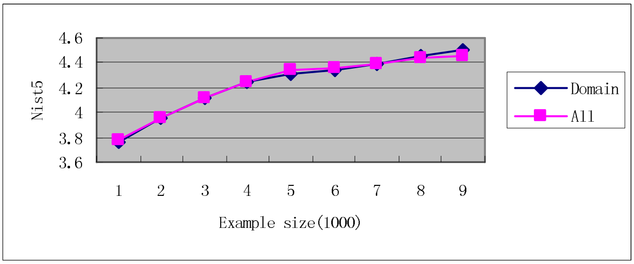


Fig. 3. EBMT Performance in Traffic Domain vs. in Whole Corpus

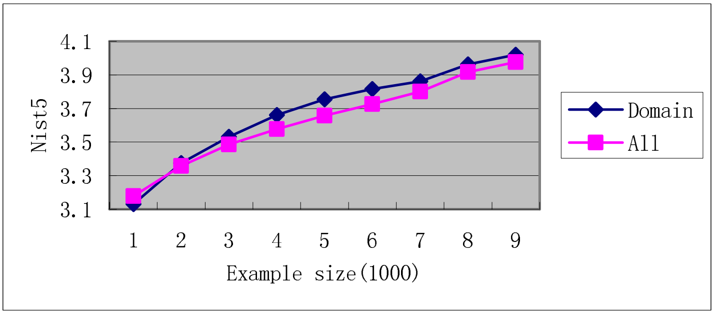


Fig. 4. EBMT Performance in Sports Domain vs. in Whole Corpus

In fact, when 5 other linguistic experts are invited to examine this corpus, they would rather interpret it like this:

1) All tourism-related texts are grouped into traveling domain, with the exception of food&drink and traffic texts.

2) The business domain actually contains two completely different types of languages: one is dialogue and the other is business letters;

3) The sports domain is clearly distinguished by conversations on sports activities.

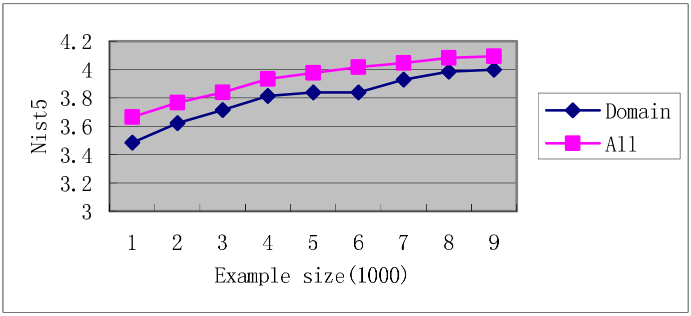


Fig. 5. EBMT Performance in Travel Domain vs. in Whole Corpus

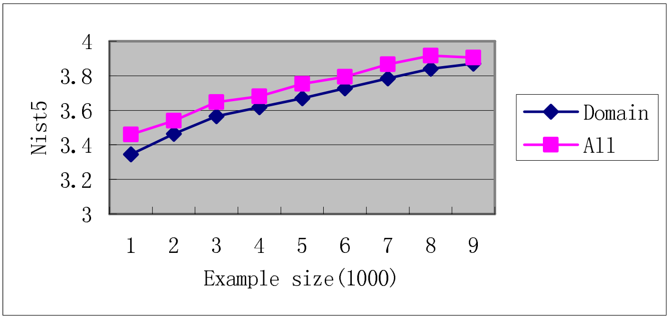


Fig. 6. EBMT Performance in Business Domain vs. in Whole Corpus

Table 1. Statistics of Current Olympic Oriented Trilingual Corpus

Domain	Number of Sentence Pair	Number of Word	Scenario Tags	Topic tags
Traveling	11,408	441,015	19	128
Food &Drink	11,352	441,018	9	15
Sports	95,09	605,663	10	47
Traffic	11,869	460,543	16	55
Business	9,905	475,987	4	78
Total	54,043	2,424,226	58	323

To make a remedy for this problem, more data are added to the domain of traveling and business. We also take out some noisy data from the 5 domains by observing abnormal system performance change. After adding Japanese translation for the corpus, we finally got the 1st stage of Olympic Oriented Trilingual Corpus, for which Table 1 lists its brief statistics.

5 Conclusion

This paper presents the construction of the Olympic Oriented Trilingual Corpus, which is featured by multi domain, tri-language and practical application needs. This may be the first attempt to construct a parallel corpus for the demands of successful development of NLP applications. Instead of decide "how much corpus is enough", this paper proposes to determine "how much should be the minimum" by so called "application driven corpus scale measure". This provides a solution to multi-domain corpus construction.

Although the ten-fold cross validation is carried to verify the soundness of the corpus scale, the statistical significance of the performance differences has not been finished. Also comparison of other NLP tasks for scale measure against MT is still under investigating.

In fact, the usefulness of the proposed measure can only be validated by the feedback from the users of the corpus. At present, the corpus annotated with the domain, location and topic has been converted into SML format, which is already available in Chinese LDC (http://www.chineseldc.org).

Acknowledgments. The research is supported by National Hi-tech 863 Peoject(No. 2004AA117010-08) and Natural Science Foundation of China (Project No. 60375019).

References

1. Huang,C., Li, J.: Corpus Linguistics. The Commercial Press, Beijing (2003)
2. Arranz, V., Castell, N. et al: Bilingual Connections for Trilingual Corpora: An XML Approach. In: Proc. of LREC 2004, Lisbon, Portugal. (2004)
3. Yang, C. C., Li, K. W.: Automatic Construction of English/Chinese Parallel Corpora. Journal of the American Society for Information Science and Technology, Vol.54, No.8 (2003)
4. Resnik, P., Smith, N. A.: The Web as a Parallel Corpus. Computational Linguistics, Vol. 29, Issue 3 (2003)
5. Yang, M., Tiejun, Z., et al: Auto Word Alignment Based Chinese-English EBMT. Proc. of IWSLT, Kyoto, Japan (2004)
6. Doddington, G. Automatic Evaluation of Machine Translation Quality using N-gram Co-Occurrence Statistics. Proc. of Human Language Technology, San Diego (2002)

The Contribution of Lexical Resources to Natural Language Processing of CJK Languages

Jack Halpern（春遍雀來）

The CJK Dictionary Institute (CJKI) (日中韓辭典研究所)
34-14, 2-chome, Tohoku, Niiza-shi, Saitama 352-0001, Japan
jack@cjk.org

Abstract. The role of lexical resources is often understated in NLP research. The complexity of Chinese, Japanese and Korean (CJK) poses special challenges to developers of NLP tools, especially in the area of word segmentation (WS), information retrieval (IR), named entity extraction (NER), and machine translation (MT). These difficulties are exacerbated by the lack of comprehensive lexical resources, especially for proper nouns, and the lack of a standardized orthography, especially in Japanese. This paper summarizes some of the major linguistic issues in the development NLP applications that are dependent on lexical resources, and discusses the central role such resources should play in enhancing the accuracy of NLP tools.

1 Introduction

Developers of CJK NLP tools face various challenges, some of the major ones being:

1. Identifying and processing the large number of orthographic variants in Japanese, and alternate character forms in CJK languages.
2. The lack of easily available comprehensive lexical resources, especially lexical databases, comparable to the major European languages.
3. The accurate conversion between Simplified and Traditional Chinese [7].
4. The morphological complexity of Japanese and Korean.
5. Accurate word segmentation ([3], [12]) and disambiguating ambiguous segmentations strings (ASS) [15].
6. The difficulty of lexeme-based retrieval and CJK CLIR [4].
7. Chinese and Japanese proper nouns, which are very numerous, are difficult to detect without a lexicon.
8. Automatic recognition of terms and their variants [6].

The various attempts to tackle these tasks by statistical and algorithmic methods [10] have had only limited success. An important motivation for such methodology has been the poor availability and high cost of acquiring and maintaining large-scale lexical databases.

This paper discusses how a lexicon-driven approach exploiting large-scale lexical databases can offer reliable solutions to some of the principal issues, based on over a decade of experience in building such databases for NLP applications.

Q. Huo et al. (Eds.): ISCSLP 2006, LNAI 4274, pp. 768 – 780, 2006.

2 Named Entity Extraction

Named Entity Recognition. (NER) is useful in NLP applications such as question answering, machine translation and information extraction. A major difficulty in NER, and a strong motivation for using tools based on probabilistic methods, is that the compilation and maintenance of large entity databases is time consuming and expensive.

The number of personal names and their variants (e.g. over a hundred ways to spell *Mohammed*) is probably in the billions. The number of place names is also large, though they are relatively stable compared with the names of organizations and products, which change frequently.

A small number of organizations, including The CJK Dictionary Institute (CJKI), maintain databases of millions of proper nouns, but even such comprehensive databases cannot be kept fully up-to-date as countless new names are created daily. Various techniques have been used to automatically detect entities, one being the use of keywords or syntactic structures that co-occur with proper nouns, which we refer to as *named entity contextual clues* (NECC).

Table 1 shows NECCs for Japanese proper nouns, which when used in conjunction with entity lexicons like the one shown in Table 2 below achieve high precision in entity recognition. Of course for NER there is no need for such lexicons to be multilingual, though it is obviously essential for MT.

Table 1. Named Entity Contextual Clues

Headword	Reading	Example
センター	せんたー	国民生活**センター**
ホテル	ほてる	ホテルシオノ
駅	えき	朝霞**駅**
協会	きょうかい	日本ユニセフ**協会**

Table 2. Multilingual Database of Place Names

English	Japanese	Simplified Chinese	LO	Traditional Chinese	Korean
Azerbaijan	アゼルバイジャン	阿塞拜疆	L	亞塞拜然	아제르바이잔
Caracas	カラカス	加拉加斯	L	卡拉卡斯	카라카스
Cairo	カイロ	开罗	O	開羅	카이로
Chad	チャド	乍得	L	查德	차드
New Zealand	ニュージーランド	新西兰	L	紐西蘭	뉴질랜드
Seoul	ソウル	首尔	O	首爾	서울
Seoul	ソウル	汉城	O	漢城	서울
Yemen	イエメン	也门	L	葉門	예멘

Note how the lexemic pairs (“L” in the **LO** column) in Table 2 above are not merely simplified and traditional *orthographic* (“O”) versions of each other, but independent lexemes equivalent to American *truck* and British *lorry.*

NER, especially of personal names and place names, is an area in which lexicon-driven methods have a clear advantage over probabilistic methods and in which the role of lexical resources should be a central one.

3 Linguistic Issues in Chinese

3.1 Processing Multiword Units

A major issue for Chinese segmentors is how to treat compound words and multiword lexical units (MWU), which are often decomposed into their components rather than treated as single units.

For example, 录像带 *lùxiàngdài* 'video cassette' and 机器翻译 *jīqifānyì* 'machine translation' are not tagged as segments in Chinese Gigaword, the largest tagged Chinese corpus in existence, processed by the CKIP morphological analyzer [13]. Possible reasons for this include:

1. The lexicons used by Chinese segmentors are small-scale or incomplete. Our testing of various Chinese segmentors has shown that coverage of MWUs is often limited.
2. Chinese linguists disagree on the concept of wordhood in Chinese. Various theories such as the Lexical Integrity Hypothesis [5] have been proposed. Packard’s outstanding book [2] on the subject clears up much of the confusion.
3. The "correct” segmentation can depend on the application, and there are various segmentation standards. For example, a search engine user looking for 录像带 is not normally interested in 录像 'to videotape' and 带 'belt' per se, unless they are part of 录像带.

This last point is important enough to merit elaboration. A user searching for 中国人 *zhōngguórén* 'Chinese (person)' is *not* interested in 中国 'China', and vice-versa. A search for 中国 should *not* retrieve 中国人 as an instance of 中国. Exactly the same logic should apply to 机器翻译, so that a search for that keyword should only retrieve documents containing that string in its entirety. Yet performing a Google search on 机器翻译 in normal mode gave some 2.3 million hits, hundreds of thousands of which had zero occurrences of 机器翻译 but numerous occurrences of unrelated words like 机器人 'robot', which the user is not interested in.

This is equivalent to saying that *headwaiter* should not be considered an instance of *waiter*, which is indeed how Google behaves. More to the point, English space-delimited lexemes like *high school* are not instances of the adjective *high*. As shown in [9], "the degree of solidity often has nothing to do with the status of a string as a lexeme. *School bus* is just as legitimate a lexeme as is *headwaiter* or *word-processor*.

The presence or absence of spaces or hyphens, that is, the orthography, does not determine the lexemic status of a string."

In a similar manner, it is perfectly legitimate to consider Chinese MWUs like those shown below as indivisible units for most applications, especially information retrieval and machine translation.

丝绸之路	*sīchóuzhīlù*	silk road
机器翻译	*jīqifānyì*	machine translation
爱国主义	*àiguózhǔyì*	patriotism
录像带	*lùxiàngdài*	video cassette
新西兰	*Xīnxīlán*	New Zealand
临阵磨枪	*línzhènmóqiāng*	start to prepare at the last moment

One could argue that 机器翻译 is compositional and therefore should be considered "two words." Whether we count it as one or two "words" is not really relevant – what matters is that it is *one lexeme* (smallest distinctive units associating meaning with form). On the other extreme, it is clear that idiomatic expressions like 临阵磨枪, literally "sharpen one's spear before going to battle," meaning 'start to prepare at the last moment,' are indivisible units.

Predicting compositionality is not trivial and often impossible. For many purposes, the only practical solution is to consider all lexemes as indivisible. Nonetheless, currently even the most advanced segmentors fail to identify such lexemes and missegment them into their constituents, no doubt because they are not registered in the lexicon. This is an area in which expanded lexical resources can significantly improve segmentation accuracy.

In conclusion, lexical items like 机器翻译 'machine translation' represent standalone, well-defined concepts and should be treated as single units. The fact that in English *machineless* is spelled solid and *machine translation* is not is an historical accident of orthography unrelated to the fundamental fact that both are full-fledged lexemes each of which represents an indivisible, independent concept. The same logic applies to 机器翻译, which is a full-fledged lexeme that should not be decomposed.

3.2 Multilevel Segmentation

Chinese MWUs can consist of nested components that can be segmented in different ways for different levels to satisfy the requirements of different segmentation standards. The example below shows how 北京日本人学校 *Běijīng Rìběnrén Xuéxiào* 'Beijing School for Japanese (nationals)' can be segmented on five different levels.

1. 北京日本人学校 multiword lexemic
2. 北京+日本人+学校 lexemic
3. 北京+日本+人+学校 sublexemic
4. 北京 + [日本 + 人] [学+校] morphemic
5. [北+京] [日+本+人] [学+校] submorphemic

For some applications, such as MT and NER, the multiword lexemic level is most appropriate (the level most commonly used in CJKI's dictionaries). For others, such as embedded speech technology where dictionary size matters, the lexemic level is best. A more advanced and expensive solution is to store presegmented MWUs in the lexicon, or even to store nesting delimiters as shown above, making it possible to select the desired segmentation level.

The problem of incorrect segmentation is especially obvious in the case of neologisms. Of course no lexical database can expect to keep up with the latest neologisms, and even the first edition of Chinese Gigaword does not yet have 博客 *bókè* 'blog'. Here are some examples of MWU neologisms, some of which are not (at least bilingually), compositional but fully qualify as lexemes.

电脑迷 *diànnǎomí* cyberphile
电子商务 *diànzǐshāngwù* e-commerce
追车族 *zhuīchēzú* auto fan

3.3 Chinese-to-Chinese Conversion (C2C)

Numerous Chinese characters underwent drastic simplifications in the postwar period. Chinese written in these simplified forms is called Simplified Chinese (SC). Taiwan, Hong Kong, and most overseas Chinese continue to use the old, complex forms, referred to as Traditional Chinese (TC). Contrary to popular perception, the process of accurately converting SC to/from TC is full of complexities and pitfalls. The linguistic issues are discussed in [7], while technical issues are described in [11]. The conversion can be implemented on three levels in increasing order of sophistication:

1. Code Conversion. The easiest, but most unreliable, way to perform C2C is to transcode by using a one-to-one mapping table. Because of the numerous one-to-many ambiguities, as shown below, the rate of conversion failure is unacceptably high.

Table 3. Code Conversion

SC	TC1	TC2	TC3	TC4	Remarks
门	們				one-to-one
汤	湯				one-to-one
发	發	髮			one-to-many
暗	暗	闇			one-to-many
干	幹	乾	干	榦	one-to-many

2. Orthographic Conversion. The next level of sophistication is to convert orthographic units, rather than codepoints. That is, meaningful linguistic units, equivalent to lexemes, with the important difference that the TC is the traditional version of the

SC on a character form level. While code conversion is ambiguous, orthographic conversion gives much better results because the orthographic mapping tables enable conversion on the lexeme level, as shown below.

Table 4. Orthographic Conversion

English	SC	TC1	TC2	Incorrect
Telephone	电话	電話		
Dry	干燥	乾燥		干燥 幹燥 榦燥
	阴干	陰乾	陰干	

As can be seen, the ambiguities inherent in code conversion are resolved by using orthographic mapping tables, which avoids false conversions such as shown in the **Incorrect** column. Because of segmentation ambiguities, such conversion must be done with a segmentor that can break the text stream into meaningful units [3].

An extra complication, among various others, is that some lexemes have one-to-many orthographic mappings, *all* of which are correct. For example, SC 阴干 correctly maps to both TC 陰乾 'dry in the shade' and TC 陰干 'the five even numbers'. Well designed orthographic mapping tables must take such anomalies into account.

3. Lexemic Conversion. The most sophisticated form of C2C conversion is called *lexemic conversion,* which maps SC and TC lexemes that are semantically, not orthographically, equivalent. For example, SC 信息 *xìnxī* 'information' is converted into the semantically equivalent TC 資訊 *zīxùn*. This is similar to the difference between British *pavement* and American *sidewalk.* [14] has demonstrated that there are numerous lexemic differences between SC and TC, especially in technical terms and proper nouns, e.g. there are more than 10 variants for *Osama bin Laden.*

Table 5. Lexemic Conversion

English	SC	Taiwan TC	HK TC	Incorrect TC
Software	软件	軟體	軟件	軟件
Taxi	出租汽车	計程車	的士	出租汽車
Osama Bin Laden	奥萨马 本拉登	奧薩瑪賓拉登	奧薩瑪 賓拉丹	奧薩馬本 拉登
Oahu	瓦胡岛	歐胡島		瓦胡島

3.4 Traditional Chinese Variants

Traditional Chinese has numerous variant character forms, leading to much confusion. Disambiguating these variants can be done by using mapping tables such as the one shown below.

If such a table is carefully constructed by limiting it to cases of 100% semantic interchangeability for polysemes, it is easy to normalize a TC text by trivially replacing variants by their standardized forms. For this to work, all relevant components, such as MT dictionaries, search engine indexes and the related documents should be normalized. An extra complication is that Taiwanese and Hong Kong variants are sometimes different [14].

Table 6. TC Variants

Var. 1	Var. 2	English	Comment
裏	裡	Inside	100% interchangeable
著	着	Particle	variant 2 not in Big5
沉	沈	sink; surname	partially interchangeable

4 Orthographic Variation in Japanese

4.1 Highly Irregular Orthography

The Japanese orthography is highly irregular, significantly more so than any other major language, including Chinese. A major factor is the complex interaction of the four scripts used to write Japanese, e.g. kanji, hiragana, katakana, and the Latin alphabet, resulting in countless words that can be written in a variety of often unpredictable ways, and the lack of a standardized orthography. For example, *toriatsukai* 'handling' can be written in six ways: 取り扱い, 取扱い, 取扱, とり扱い, 取りあつかい, とりあつかい.

An example of how difficult Japanese IR can be is the proverbial 'A hen that lays golden eggs.' The "standard" orthography would be 金の卵を産む鶏 *Kin no tamago o umu niwatori*. In reality, *tamago* 'egg' has four variants (卵, 玉子, たまご, タマゴ), *niwatori* 'chicken' three (鶏, にわとり, ニワトリ) and *umu* 'to lay' two (産む, 生む), which expands to 24 permutations like 金の卵を生むニワトリ, 金の玉子を産む鶏 etc. As can be easily verified by searching the web, these variants occur frequently.

Linguistic tools that perform segmentation, MT, entity extraction and the like must identify and/or normalize such variants to perform dictionary lookup. Below is a brief discussion of what kind of variation occurs and how such normalization can be achieved.

4.2 Okurigana Variants

One of the most common types of orthographic variation in Japanese occurs in kana endings, called *okurigana*, that are attached to a kanji stem. For example, *okonau* 'perform' can be written 行うor 行なう, whereas *toriatsukai* can be written in the six ways shown above. Okurigana variants are numerous and unpredictable. Identifying them must play a major role in Japanese orthographic normalization. Although it is possible to create a dictionary of okurigana variants algorithmically, the resulting lexicon would be huge and may create numerous false positives not semantically interchangeable. The most effective solution is to use a lexicon of okurigana variants, such as the one shown below:

Table 7. Okurigana Variants

HEADWORD	READING	NORMALIZED
書き著す	かきあらわす	書き著す
書き著わす	かきあらわす	書き著す
書著す	かきあらわす	書き著す
書著わす	かきあらわす	書き著す

Since Japanese is highly agglutinative and verbs can have numerous inflected forms, a lexicon such as the above must be used in conjunction with a morphological analyzer that can do accurate stemming, i.e. be capable of recognizing that 書き著しませんでした is the polite form of the canonical form 書き著す.

4.3 Cross-Script Orthographic Variation

Variation across the four scripts in Japanese is common and unpredictable, so that the same word can be written in any of several scripts, or even as a hybrid of multiple scripts, as shown below:

Table 8. Cross-Script Variation

Kanji	Hiragana	katakana	Latin	Hybrid	Gloss
人参	にんじん	ニンジン			carrot
		オープン	OPEN		open
硫黄		イオウ			sulfur
		ワイシャツ		Yシャツ	shirt
皮膚		ヒフ		皮フ	skin

Cross-script variation can have major consequences for recall, as can be seen from the table below.

Table 9. Hit Distribution for 人参 'carrot' *ninjin*

ID	Keyword	Normalized	Google Hits
A	人参	人参	67,500
B	にんじん	人参	66,200
C	ニンジン	人参	58,000

Using the ID above to represent the number of Google hits, this gives a total of A＋B＋C＋α_{123} = 191,700. α is a coincidental occurrence factor, such as in '100人参加, in which '人参' is unrelated to the 'carrot' sense. The formulae for calculating the above are as follows.

Unnormalized recall:

$$\frac{C}{A+B+C+\alpha_{123}} = \frac{58.000}{191.700} \quad (\approx 30\%) \tag{1}$$

Normalized recall:

$$\frac{A+B+C}{A+B+C+\ \alpha_{123}} = \frac{191.700}{191.700} \quad (\approx 100\%) \tag{2}$$

Unnormalized precision:

$$\frac{C}{C+\alpha_{3}} = \frac{58.000}{58.000} \quad (\approx 100\%) \tag{3}$$

Normalized precision:

$$\frac{C}{A+B+C+\alpha_{123}} = \frac{191.000}{191.000} \quad (\approx 100\%) \tag{4}$$

人參 'carrot' illustrates how serious a problem cross-orthographic variants can be. If orthographic normalization is not implemented to ensure that all variants are indexed on a standardized form like 人參, recall is only 30%; if it is, there is a dramatic improvement and recall goes up to nearly 100%, without any loss in precision, which hovers at 100%.

4.4 Kana Variants

A sharp increase in the use of katakana in recent years is a major annoyance to NLP applications because katakana orthography is often irregular; it is quite common for the same word to be written in multiple, unpredictable ways. Although hiragana orthography is generally regular, a small number of irregularities persist. Some of the major types of kana variation are shown in the table below.

Table 10. Kana Variants

Type	English	Standard	Variants
Macron	computer	コンピュータ	コンピューター
Long vowels	maid	メード	メイド
Multiple kana	team	チーム	ティーム
Traditional	big	おおきい	おうきい
づ vs. ず	continue	つづく	つずく

The above is only a brief introduction to the most important types of kana variation. Though attempts at algorithmic solutions have been made by some NLP research laboratories [1], the most practical solution is to use a katakana normalization table, such as the one shown below, as is being done by Yahoo! Japan and other major portals.

Table 11. Kana Variants

HEADWORD	NORMALIZED	English
アーキテクチャ	アーキテクチャー	Architecture
アーキテクチャー	アーキテクチャー	Architecture
アーキテクチュア	アーキテクチャー	Architecture

4.5 Miscellaneous Variants

There are various other types of orthographic variants in Japanese, described in [8]. To mention some, kanji even in contemporary Japanese sometimes have variants, such as 才 for 歳 and 巾 for 幅, and traditional forms such as 發 for 発. In addition, many *kun* homophones and their variable orthography are often close or even identical in meaning, i.e., *noboru* means 'go up' when written 上る but 'climb' when written 登る, so that great care must be taken in the normalization process so as to assure semantic interchangeability for all senses of polysemes; that is, to ensure that such forms are *excluded* from the normalization table.

Table 12. Orthographic Normalization Table

HEADWORD	READING	NORMALIZED
空き缶	あきかん	空き缶
空缶	あきかん	空き缶
明き罐	あきかん	空き缶
あき缶	あきかん	空き缶
あき罐	あきかん	空き缶
空きかん	あきかん	空き缶
空きカン	あきかん	空き缶
空き罐	あきかん	空き缶
空罐	あきかん	空き缶
空き鑵	あきかん	空き缶
空鑵	あきかん	空き缶

4.6 Lexicon-Driven Normalization

Leaving statistical methods aside, lexicon- driven normalization of Japanese orthographic variants can be achieved by using an orthographic mapping table such as the one shown below, using various techniques such as:

1. Convert variants to a standardized form for indexing.
2. Normalize queries for dictionary lookup.
3. Normalize all source documents.
4. Identify forms as members of a variant group.

Other possibilities for normalization include advanced applications such as domain-specific synonym expansion, requiring Japanese thesauri based on domain ontologies, as is done by a select number of companies like Wand and Convera who build sophisticated Japanese IR systems.

5 Orthographic Variation in Korean

Modern Korean has is a significant amount of orthographic variation, though far less than in Japanese. Combined with the morphological complexity of the language, this poses various challenges to developers of NLP tools. The issues are similar to Japanese in principle but differ in detail.

Briefly, Korean has variant hangul spellings in the writing of loanwords, such as 케이크 *keikeu* and 케익 *keik* for 'cake', and in the writing of non-Korean personal names, such as 클린턴 *keulrinteon* and 클린톤*keulrinton* for 'Clinton'. In addition, similar to Japanese but on a smaller scale, Korean is written in a mixture of hangul, Chinese characters and the Latin alphabet. For example, 'shirt' can be written 와이셔츠 *wai-syeacheu* or Y셔츠 *wai-syeacheu*, whereas 'one o'clock' *hanzi* can written as 한시, 1시 or 一時. Another issue is the differences between South and North Korea spellings, such as N.K. 오사까 *osakka* vs. S.K. 오사카 *osaka* for 'Osaka', and the old (pre-1988) orthography versus the new, i.e. modern 일군 'worker' (*ilgun*) used to be written 일꾼 (*ilkkun*).

Lexical databases, such as normalization tables similar to the ones shown above for Japanese, are the only practical solution to identifying such variants, as they are in principle unpredictable.

6 The Role of Lexical Databases

Because of the irregular orthography of CJK languages, procedures such as orthographic normalization cannot be based on statistical and probabilistic methods (e.g. bigramming) alone, not to speak of pure algorithmic methods. Many attempts have been made along these lines, as for example [1] and [4], with some claiming performance equivalent to lexicon-driven methods, while [10] reports good results with only a small lexicon and simple segmentor.

[3] and others have reported that a robust morphological analyzer capable of processing lexemes, rather than bigrams or n-grams, must be supported by a large-scale computational lexicon. This experience is shared by many of the world's major portals and MT developers, who make extensive use of lexical databases.

Unlike in the past, disk storage is no longer a major issue. Many researchers and developers, such as Prof. Franz Guenthner of the University of Munich, have come to realize that "language is in the data," and "the data is in the dictionary," even to the point of compiling full-form dictionaries with millions of entries rather than rely on statistical methods, such as Meaningful Machines who use a full form dictionary containing millions of entries in developing a human quality Spanish-to-English MT system.

CJKI, which specializes in CJK and Arabic computational lexicography, is engaged in an ongoing research and development effort to compile CJK and Arabic lexical databases (currently about seven million entries), with special emphasis on proper nouns, orthographic normalization, and C2C. These resources are being subjected to heavy industrial use under real-world conditions, and the feedback thereof is being used to further expand these databases and to enhance the effectiveness of the NLP tools based on them.

7 Conclusions

Performing such tasks as orthographic normalization and named entity extraction accurately is beyond the ability of statistical methods alone, not to speak of C2C conversion and morphological analysis. However, the small-scale lexical resources currently used by many NLP tools are inadequate to these tasks. Because of the irregular orthography of the CJK writing systems, lexical databases fine-tuned to the needs of NLP applications are required. The building of large-scale lexicons based on corpora consisting of even billions of words has come of age. Since lexicon-driven techniques have proven their effectiveness, there is no need to overly rely on probabilistic methods. Comprehensive, up-to-date lexical resources are the key to achieving major enhancements in NLP technology.

References

1. Brill, E. and Kacmarick, G. and Brocket, C. (2001) *Automatically Harvesting Katakana-English Term Pairs from Search Engine Query Logs*. Microsoft Research, Proc. of the Sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan.
2. Packard, L. Jerome (1998) "New Approaches to Chinese Word Formation", Mouton Degruyter, Berlin and New York.
3. Emerson, T. (2000) Segmenting Chinese in Unicode. Proc. of the 16th International Unicode Conference, Amsterdam
4. Goto, I., Uratani, N. and Ehara T. (2001) *Cross-Language Information Retrieval of Proper Nouns using Context Information*. NHK Science and Technical Research Laboratories. Proc. of the Sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan
5. Huang, James C. (1984) *Phrase Structure, Lexical Integrity, and Chinese Compounds,* Journal of the Chinese Teachers Language Association, 19.2: 53-78
6. Jacquemin, C. (2001) Spotting and Discovering Terms through Natural Language Processing. The MIT Press, Cambridge, MA
7. Halpern, J. and Kerman J. (1999) *The Pitfalls and Complexities of Chinese to Chinese Conversion*. Proc. of the Fourteenth International Unicode Conference in Cambridge, MA.

8. Halpern, J. (2000a) *The Challenges of Intelligent Japanese Searching*. Working paper (www.cjk.org/cjk/joa/joapaper.htm), The CJK Dictionary Institute, Saitama, Japan.
9. Halpern, J. (2000b) *Is English Segmentation Trivial?*. Working paper, (www.cjk.org/cjk/reference/engmorph.htm) The CJK Dictionary Institute, Saitama, Japan.
10. Kwok, K.L. (1997) *Lexicon Effects on Chinese Information Retrieval*. Proc. of 2nd Conf. on Empirical Methods in NLP. ACL. pp.141-8.
11. Lunde, Ken (1999) *CJKV Information Processing*. O'Reilly & Associates, Sebastopol, CA.
12. Yu, Shiwen, Zhu, Xue-feng and Wang, Hui (2000) *New Progress of the Grammatical Knowledge-base of Contemporary Chinese*. Journal of Chinese Information Processing, Institute of Computational Linguistics, Peking University, Vol.15 No.1.
13. Ma, Wei-yun and Chen, Keh-Jiann (2003) *Introduction to CKIP Chinese Word Segmentation System for the First International Chinese Word Segmentation Bakeoff,* Proceedings of the Second SIGHAN Workshop on Chinese Language Processing pp. 168-171 Sapporo, Japan
14. Tsou, B.K., Tsoi, W.F., Lai, T.B.Y. Hu, J., and Chan S.W.K. (2000) *LIVAC, a Chinese synchronous corpus, and some applications*. In "2000 International Conference on Chinese Language ComputingICCLC2000", Chicago.
15. Zhou, Qiang. and Yu, Shiwen (1994) *Blending Segmentation with Tagging in Chinese Language Corpus Processing,* 15th International Conference on Computational Linguistics (COLING 1994).

Multilingual Spoken Language Corpus Development for Communication Research

Toshiyuki Takezawa

National Institute of Information and Communications Technology
ATR Spoken Language Communication Research Laboratories
2-2-2 Hikaridai, Keihanna Science City, Kyoto 619-0288, Japan
toshiyuki.takezawa@{nict.go.jp, atr.jp}

Abstract. A multilingual spoken language corpus is indispensable for spoken language communication research such as speech-to-speech translation. To promote multilingual spoken language research and development, unified structure and annotation, such as tagging, is indispensable for both speech and natural language processing. We describe our experience with multilingual spoken language corpus development at our research institution, focusing in particular on speech recognition and natural language processing for speech translation of travel conversations.

1 Introduction

Various kinds of corpora developed for analysis of linguistic phenomena and statistical information gathering are now accessible via electronic media and can be utilized for the study of natural language processing. Since such information includes written-language corpora and monolingual corpora, however, it is not necessarily useful for research and development of multilingual spoken language processing. A multilingual spoken language corpus is indispensable for spoken language communication research such as speech-to-speech translation.

There are a variety of requirements for every component technology, such as speech recognition and language processing. A variety of speakers and pronunciations might be important for speech recognition. A variety of expressions and information on parts of speech might be important for natural language processing. To promote multilingual spoken language research and development, unified structure and annotation, such as tagging, is indispensable for both speech and natural language processing.

We describe our experience of multilingual spoken language corpus development at our research institution, focusing in particular on speech recognition and natural language processing for speech translation of travel conversations.

First, we introduce an interpreter-aided multilingual spoken dialog corpus (SLDB), and discuss corpus configuration. Next, we introduce our basic travel expression corpus (BTEC) built for training machine translation of spoken language among Japanese, English, and Chinese. Finally, we discuss the Japanese, English, and Chinese multilingual spoken dialog corpus that we created using speech-to-speech translation systems.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 781–791, 2006.

2 Interpreter-Aided Multilingual Spoken Dialog Corpus (SLDB)

The corpus, called Spoken Language DataBase (SLDB), contains data from dialog spoken between English and Japanese speakers through human interpreters [1,2]. All utterances in the SLDB have been translated into Chinese. The content is all travel conversations between a foreign tourist and a front desk clerk at a hotel. The interpreters serve as the speech translation system.

Table 1 is an overview of the corpus, and Table 2 shows its basic characteristics.

Table 1. Overview of SLDB

Number of collected dialogs	618
Speaker participants	71
Interpreter participants	23

Table 2. Basic characteristics of SLDB

	Japanese	English
Number of utterances	16,084	16,084
Number of sentences	21,769	22,928
Number of word tokens	236,066	181,263
Number of word types	5,298	4,320
Average number of words per sentence	10.84	7.91

Corpus configuration. One remarkable characteristic of SLDB is its integration of speech and linguistic data. Each conversation includes recorded speech data, transcribed utterances, and their correspondences.

The transcribed Japanese and English utterances are tagged with morphological and syntactic information. These kinds of tagged information are crucial for natural language processing. The recorded speech signals and transcribed utterances in our database provide us with both examples of various phenomena in bilingual conversations, and input data for speech recognition and machine translation evaluation purposes.

Data can be classified into the following three major categories.

1. Transcribed data
2. Tagged data
3. Speech data

The transcribed data consists of the following.

(a) Bilingual text
(b) Japanese text
(c) English text

```
J: Arigatou gozai masu. Kyoto Kankou Hotel de gozai masu.
JE: Thank you for calling Kyoto Kanko Hotel.|How may I help you?

E: Good evening. |I'd like to make a reservation, please.
EJ: Kombanwa. |Yoyaku wo shi tai n desu keredomo.

J: Hai,[e-]go yoyaku no hou wa itsu desho u ka?
JE: Yes, when do you plan to stay?

E: I'd like to stay from August tenth through the twelfth, for two nights.|
   If possible, I'd like a single room, please.
EJ: Hachigatsu no tooka kara juuni-nichi made, ni-haku shi tai n desu.|
    Dekire ba, single room de o-negai shi masu.

J: Kashikomari mashi ta. |Shoushou o-machi kudasai mase.
JE: All right, please wait a moment.
J: O-mata se itashi mashi ta.|
   Osoreiri masu ga, single room wa manshitsu to nat te ori masu.
JE: I am very sorry our single rooms are fully booked.
J: [e]Washitsu ka twin room no o-hitori sama shiyou deshi tara o-tori deki masu ga.
JE: But, Japanese style rooms and twin rooms for single use are available.

E: [Oh] what are the rates on those types of rooms?
EJ: Sono o-heya no ryoukin wo oshie te kudasai.

J: Hai,[e-]twin room no o-hitori sama shiyou desu to ichi-man yon-sen yen.
JE: Yes, a twin room for single use is fourteen thousand yen.
J: Washitsu no hou desu to ichi-man has-sen yen,
  [e-] izuremo zei, service ryou wa betsu ni nari masu.
JE: And for a Japanese style room, it's eighteen thousand yen per night,
    and tax and service charges are not included.
```

Fig. 1. Conversation between an American tourist and a Japanese front desk clerk

The recorded bilingual conversations are transcribed into a text file. Bilingual text contains descriptions of the situations in which a speech translation system is utilized.

Figure 1 shows examples of transcribed conversations. The Japanese text in Figure 1 is transcribed into Romanized Japanese for the convenience of readers who are unfamiliar with Japanese *hiragana*, *katakana*, and *kanji* (Chinese characters). The original text is transcribed in Japanese characters *hiragana*, *katakana*, and *kanji*. Interjections are bracketed. J, E, JE, or EJ at the beginning of a line denotes a Japanese speaker, an English speaker, a Japanese-to-English interpreter, or an English-to-Japanese interpreter, respectively. "|" denotes a sentence boundary. A blank line between utterances shows that the utterance's right was transferred.

The Japanese text is produced by extracting the utterances of a Japanese speaker and an English-to-Japanese interpreter, while the English text is produced by extracting the utterances of an English speaker and a Japanese-to-English interpreter. These two kinds of data are utilized for such monolingual investigations as morphological analysis, parsing, and so on.

The tagged data consists of the following.

(d) Japanese morphological data
(e) English morphological data

Morphological and syntactic information are useful for the study of statistical natural language processing, the production of grammar rules, etc.

3 Basic Travel Expression Corpus (BTEC)

The Basic Travel Expression Corpus (BTEC) [3,4] was designed to cover utterances for all potential topics in travel conversations, together with their translations. Since it is practically impossible to collect them by transcribing actual conversations or simulated dialogs, we decided to use sentences from the memories of bilingual travel experts. We started by looking at phrasebooks that contain bilingual (in our case Japanese/English) sentence pairs that the authors consider useful for tourists traveling abroad. We collected these sentence pairs and rewrote them to make translations as context-independent as possible and to comply with our speech transcription style. Sentences outside of the travel domain or those containing very special meanings were removed.

Table 3 contains basic statistics of the BTEC collections, called BTEC1, 2, 3, 4, and 5. Each collection was created using the same procedure in a different time period. We used a morpheme as the basic linguistic unit for Japanese (instead of a word), since morpheme units are more stable than word units.

BTEC sentences, as described above, did not come from actual conversations but were generated by experts as reference materials. This approach enabled us to efficiently create a broad coverage corpus, but it may have two problems. First, this corpus may lack utterances that appear in real conversation. For example, when people ask the way to a bus stop, they often use a sentence like (1). However, BTEC1 contains (2) instead of (1).

(1) I'd like to go downtown. Where can I catch a bus?
(2) Where is a bus stop (to go downtown)?

The second problem is that the frequency distribution of this corpus may be different from the "actual" one. In this corpus, the frequency of an utterance most likely reflects the best trade-off between usefulness in real situations and compactness of the collection. Therefore, it is possible to think of this frequency distribution as a first approximation of reality, but this is an open question.

Table 3. Overview of BTEC

	BTEC1	BTEC2	BTEC3	BTEC4	BTEC5
# of utterances (10^3)	172	46	198	74	98
# of Japanese word tokens (10^3)	1,174	341	1,434	548	1,046
# of Japanese word types (10^3)	28	20	43	22	28
languages	J:EC	J:EC	J:EC	E:JC	E:JC

4 Machine-Translation-Aided Dialogs in a Laboratory Room (MAD)

The previous approach focuses on maximizing the coverage of the corpus rather than creating an accurate sample of reality. Users may use different wording when they speak to the system.

Therefore, the second approach is intended to collect representative utterances that people will input into S2ST systems. For this purpose, we carried out simulated (i.e., role play) dialogs between two native speakers of different mother tongues with a Japanese/English bi-directional S2ST system, instead of using human interpreters.

The first half period of the research program, we used human typists instead of speech recognizers in order to collect good quality data. The second half period of the research program, we used our S2ST system between English and Japanese and between Chinese and Japanese.

4.1 Collecting Spoken Dialog Data Using Typists

We have conducted five sets of simulated dialogs (MAD1 through MAD5) so far, changing parameters including system configurations, complexity of dialog tasks, instructions to speakers, etc. Table 4 shows a summary of the five experiments, MAD1-MAD5. In this table, the number of utterances includes both Japanese and English.

Table 4. Statistics of MAD Corpora

Subset ID	MAD1	MAD2	MAD3	MAD4	MAD5
Reference	[5]	[5]	[6]	[7]	[8]
# of utterances	3022	1696	2180	1872	1437
# of morphs per utterance	10.0	12.6	11.1	9.82	8.47
# of utterances per dialog	7.8	49.3	18.8	22.0	27.0
Task complexity	Simple	Complex	Medium	Medium	Medium

Average numbers depend on experimental conditions.

The first set of dialogs (MAD1) was collected to see whether conversation through a machine translation system is feasible. The second set (MAD2) focused on task achievement by assigning complex tasks to participants. The third set (MAD3) contains carefully recorded speech data of medium complexity. MAD4 and MAD5 aim to investigate how utterances change based on different settings.

Figure 2 is an overview diagram of the data collection environment.

It is very likely that people speak differently to a spoken language system based on the instructions given to them. For all the sets except MAD1, we made instructional movies to ensure that the same instructions were given to each subject. Before starting the experiments, subjects were asked to watch these movies and then to try the system with test dialogs. Instructions and practice

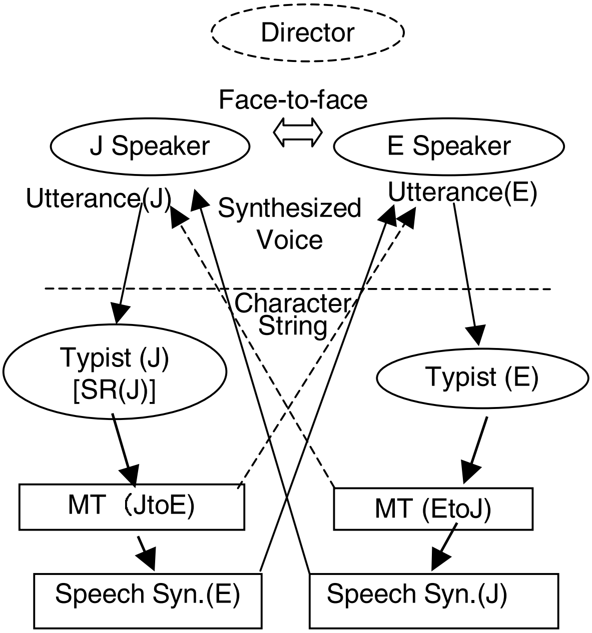


Fig. 2. Data Collection Environment of MAD

took about 30 minutes. In the fourth set (MAD4), we gave different types of instructions in this preparation step.

S2ST presupposes that each user understands the translated utterances of the other. However, the dialog environment described so far allows the user to access other information, such as translated text displayed on a PDA. We tried to control the extra information in MAD5 to see how utterances would be affected.

4.2 Collecting Spoken Dialog Data Using Speech Translation Systems

We collected spoken dialog data using our S2ST system for English and Japanese. This data collection experiment is called MAD6 because we conducted five data collection experiments using typists. The system was configure as follows.

- Acoustic model for Japanese speech recognition: Speaker-adapted models.
- Language model for Japanese speech recognition: Vocabulary size 52,000 morphemes.
- Acoustic model for English speech recognition: Speaker-adapted models.
- Language model for English speech recognition: Vocabulary size 15,000 morphemes.
- Translation from Japanese to English: HPAT+D3+Selector [9].
- Translation from English to Japanese: HPAT+D3+Selector [9].

Table 5. Overview of MAD6

	MAD6
Purpose	Spoken dialog data collection using S2ST system
Task	Simple as MAD1
# of utterances	2,507
# of dialogs	139

- Japanese speech synthesis: XIMERA 1.0 [10].
- English speech synthesis: AT&T Labs' Natural Voices^{TM}.

Table 5 is an overview of MAD6. We designed task dialogs to take ten minutes or less.

5 Machine-Translation-Aided Dialogs in Realistic Fields (FED)

An ideal approach to applying a system to "real" utterances is to let people use the system in real world settings to achieve real conversational goals (e.g., booking a package tour). This approach, however, has at least two problems. First, it is difficult to back up the system when it makes errors because current technology is not perfect. Second, it is difficult to control tasks and conditions to do meaningful analysis of the collected data.

The new experiment reported here was still in the role-play style but its dialog situations were designed to be more natural. We set up our S2ST system for travel conversation at tourist information centers in an airport and a train station, and then asked non-Japanese-speaking people to talk with the Japanese staff at information centers using the S2ST system.

Experimental system for data collection. Figure 3 is an overall diagram of the experimental system. The system includes two PDAs, one for each language, and several PC servers. The PC servers are for a special controller called the "gateway" and for component engines, consisting of ASR (Automatic Speech Recognition) [11], MT (Machine Translation) [12], and SS (Speech Synthesis) [10] PCs for each language and each language-pair. The gateway is responsible for controlling information flow between PDAs and engines. It is also responsible for mediating messages from ASR and MT engines to PDAs. Each PDA is connected to the gateway with a wireless LAN. The gateway and component engines are wired. In the FED experiment, we also used headset microphones.

An utterance spoken into a PDA is sent to the gateway server, which calls the ASR, MT, and SS engines in this order to have the utterance translated. Finally, the gateway sends the translated utterance to the other PDA.

We used speaker-adapted acoustic models for Japanese speech recognition because a limited number of Japanese staffs at the tourist office joined the FED experiment. We also added to the lexicons some proper names that were deemed

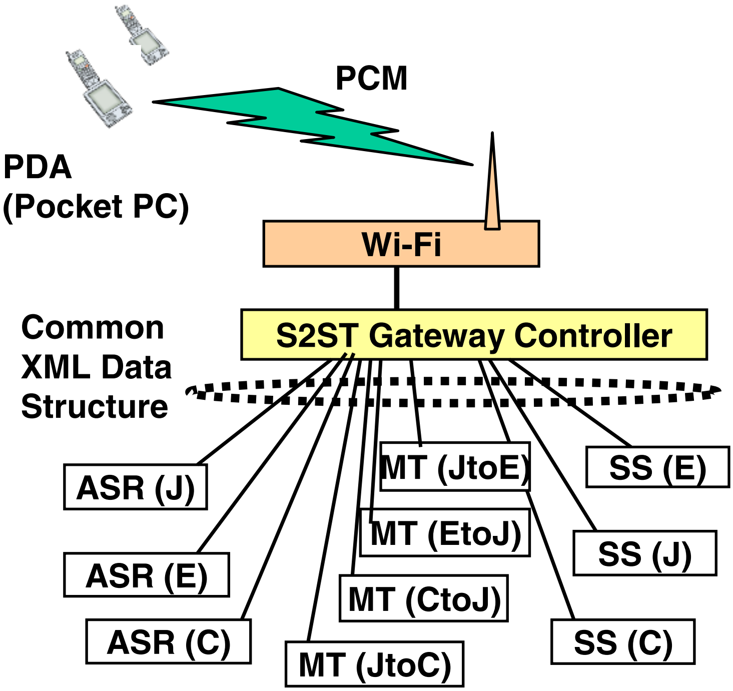


Fig. 3. Overview of the experimental system

necessary to carry out the planned conversations. These included names such as those of stations near the locations of the experiment.

Locations. We conducted the data collection experiments near two tourist information centers. One was in Kansai International Airport (hereafter, KIX), and the other was at Osaka City Air Terminal (hereafter, OCAT) in the center of Osaka. The former is in the main arrival lobby of the airport, which many tourists with luggage carts pass as they emerge from customs. The latter is a semi-enclosed area of about 40 m^2 surrounded by glass walls (but with two open doors).

Environmental noise was 60-65 dBA at both places. The noise, however, rose to 70 dBA when the public address system was in use.

Language pairs. English-Japanese/Japanese-English and Chinese-Japanese/Japanese-Chinese.

Scenario. A good method of collecting real utterances is to just let subjects talk freely without using predetermined scenarios. Analyzing uncontrolled dialog, however, is very difficult. In our FED experiment, we prepared eight dialog scenarios that were shown to subjects. These scenarios, listed below, are categorized by expected number of turns for each speaker, into three levels of complexity.

Level-1 : Requires one or two turns per speaker plus greetings.
E.g., "Please ask where the bus stop for Kyoto station is."
Level-2 : Requires three or four turns per speaker plus greetings.
E.g., "Please ask the way to Kyoto station."
Level-3 : Free discussion.
E.g., "Please ask anything related to traveling in the Osaka area."

Real dialogs included many clarification sub-dialogs necessitated by incomprehensible outputs from the system. This means that the number of turns was actually larger than we expected or planned.

Japanese speakers. We asked staff at tourist information centers to participate in the experiments, with six people at KIX and three at OCAT agreeing to take part in the experiments.

Chinese speakers. Since the Chinese speech recognizer was trained on Mandarin speech, we needed to recruit subjects from (the Beijing region of) Mainland China. It was, however, difficult to find tourists from Mainland China who had time to participate in the experiment because most of them came to Osaka as members of tightly scheduled group tours. Therefore, we relied on 36 subjects gathered by the Osaka prefectural government. These subjects are college students from Mainland China majoring in non-technical areas such as foreign studies and tourism.

English speakers. The English speech recognizer was trained on North American English. Again, however, it was difficult to find volunteer subjects who speak North American English. We expected to recruit many individual tourists, and most of the English-speaking volunteer subjects were indeed tourists arriving at or leaving the airport during the experiment. In addition to these volunteers, Osaka prefecture provided five subjects who were working in Japan as English teachers. The resulting 37 subjects were not all North Americans, as shown in Table 6.

Conducting data collection. First, we set up the S2ST system and asked the Japanese subjects (i.e., service personnel at the tourist information centers) to stand by at the experiment sites.

Table 6. Origin of English-speaking subjects

Origin	# of subjects
U.S.A.	15
GB	6
Australia	5
Canada	4
New Zealand	2
Denmark	2
Other	5

When an English or Chinese speaking subject visited a center, he or she was asked to fill out the registration form. Then, our staff explained for 2-3 minutes how to use the S2ST system and asked the subject to try very simple utterances like “hello” or “thank you.” After the trial utterances, we had the subject try two dialogs: one dialog for practice using a level-1 scenario, and the other for the “main” dialog, which was a scenario chosen randomly from level-1 through level-3. Finally, the subject was asked to answer a questionnaire.

The average time from registration to filling out the questionnaire was 15-20 minutes. Since we conducted 4-5 hours of experiments each day, excluding system setup, we were able to obtain dialog data for 15 subjects per day.

Overview of collected data. Table 7 is an overview of FED data.

Table 7. Overview of FED

	J(toE)	E(toJ)	J(toC)	C(toJ)
# of utterances	608	660	344	484
# of speakers	7	39	6	36
# of word tokens	3,851	4,306	2,017	422
# of word types	727	668	436	382

6 Conclusion

We described our experience of multilingual spoken language corpus development at our research institution, focusing in particular on speech recognition and natural language processing for speech translation of travel conversations.

First, we introduced interpreter-aided multilingual spoken dialog corpus (SLDB), and mentioned corpus configuration. Next, we introduced basic travel expression corpus (BTEC) built for training machine translation of spoken language among Japanese, English and Chinese speakers. Finally, we mentioned a multilingual spoken dialog corpus between Japanese, English, and Chinese created using speech-to-speech translation systems.

In the future, we plan to expand our activities to multilingual spoken language communication research and development involving both verbal and nonverbal communications.

Acknowledgments

This study was supported in part by the Ministry of Internal Affairs and Communications (MIC). We collected our data within the framework of “Social experiments on supporting non-Japanese-speaking tourists using information technologies”, which was carried out by the Osaka prefectural government and related

offices in winter of 2004. In these experiments, Osaka's prefectural government negotiated with management of facilities frequented by foreign tourists, such as airports and bus terminals, to provide the necessary assistance (e.g., using of public spaces and electricity). The government also gathered volunteer subjects.

References

1. Morimoto, T., Uratani, N., Takezawa, T., Furuse, O., Sobashima, Y., Iida, H., Nakamura, A., Sagisaka, Y., Higuchi, N., Yamazaki, Y.: A speech and language database for speech translation research. Proc. ICSLP (1994) 1791–1794.
2. Takezawa, T., Morimoto, T., Sagisaka, Y.: Speech and language databased for speech translation research in ATR. Proc. Oriental COCOSDA Workshop (1998) 148–155.
3. Takezawa, T., Sumita, E., Sugaya, F., Yamamoto, H., Yamamoto, S.: Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world. Proc. LREC (2002) 147–152.
4. Kikui, G., Sumita, E., Takezawa, T., Yamamoto, S.: Creating corpora for speech-to-speech translation. Proc. EUROSPEECH (2003) 381–382.
5. Takezawa, T., Kikui, G.: Collecting machine-translation-aided bilingual dialogues for corpus-based speech translation. Proc. EUROSPEECH, (2003) 2757–2760.
6. Takezawa, T., Nishino, A., Takashima, K., Matsui, T., Kikui, G.: An experimental system for collecting machine-translation-aided dialogues. Proc. Forum on Information Technology (2003) E-036.
7. Takezawa, T., Kikui, G.: A comparative study on human communication behaviors and linguistics characteristics for speech-to-speech translation. Proc. LREC (2004) 1589–1592.
8. Mizushima, M., Takezawa, T., Kikui, G.: Effects of audibility of partner's voice and visibility of translated text in machine-translation-aided bilingual spoken dialogues. IPSJ SIG Technical Reports (2004) 2004-HI-109-19/2004-SLP-52-19.
9. Sumita, E.: Example-based machine translation using DP-matching between word sequences. Proc. ACL-2001 Workshop on Data-Driven Methods in Machine Translation (2001) 1–8.
10. Kawai, H., Toda, T., Ni, J., Tsuzaki, M., Tokuda, K.: XIMERA: A new TTS from ATR based on corpus-based technologies. Proc. 5th ISCA Speech Synthesis Workshop (2004) 179–184.
11. Itoh, G., Ashikari, Y., Jitsuhiro, T., Nakamura, S.: Summary and evaluation of speech recognition integrated environment ATRASR. Autumn Meeting of the Acoustical Society of Japan (2004) 1-P-30.
12. Sumita, E., Nakaiwa, H., Yamamoto, S.: Corpus-based translation technology for multi-lingual speech-to-speech translation. Spring Meeting of the Acoustical Society of Japan (2004) 1-8-26.

Development of Multi-lingual Spoken Corpora of Indian Languages

K. Samudravijaya

Tata Institute of Fundamental Research, Homi Bhabha Road, Mumbai 400005, India
chief@tifr.res.in

Abstract. This paper describes a recently initiated effort for collection and transcription of read as well as spontaneous speech data in four Indian languages. The completed preparatory work include the design of phonetically rich sentences, data acquisition setup for recording speech data over telephone channel, a Wizard of Oz setup for acquiring speech data of a spoken dialogue of a caller with the machine in the context of a remote information retrieval task. An account of care taken to collect speech data that is as close to real world as possible is given. The current status of the programme and the set of actions planned to achieve the goal is given.

1 Introduction

Human Computer Interaction through Natural Language is touted to be the next big event in computing. This necessitates the advent of machine interfaces through which users can interact through spoken or written language. This is imperative in a multi-lingual country such as India where a large fraction of population would hesitate to use English language oriented input/output devices. In addition, development of multi-lingual technology in the Indian context would foster greater interaction among people who may not know each other's language.

Realizing the importance of multilingual technology, the Government of India has initiated a number of programmes. The most recent one is an an Indo-German Collaboration project titled 'Voice-based Multilingual Information Access (V-MIA) System for Indian Languages' [1]. Text as well as spoken multilingual corpora are key raw materials for development of language technology. This paper describes an effort for development of spoken corpora in four Indian Languages.

The organization of the paper is as follows. Section 2 briefly states the current status of corpora development in Indian languages. The goal of the current initiative is stated in Section 3. The process of accumulation of text corpus and design of phonetically rich sentence corpus is described in Section 4. The need for collecting conversational speech data for developing practical speech applications is dealt with in Section 5. A detailed account of speech data acquisition setup is given in Section 6. It also states the current status of the programme and tasks to be taken up next. Some conclusions are drawn in Section 7.

Q. Huo et al.(Eds.): ISCSLP 2006, LNAI 4274, pp. 792–801, 2006.

2 Current Scene in India

Considerable progress has been made in generation of text corpus of Indian languages thanks to the Technology Development in Indian Languages programme of the Government of India. For example, thirty lakh words of machine readable corpora in various Indian languages has been created through a dozen Language Technology Resource Centres [2]. In addition, software tools for word level tagging of grammatical categories, word count, frequency count have also been developed. In contrast, development of spoken corpora and handwritten script corpora has lagged behind. A status report of the speech database development activities for Indian languages is given in [3]. Several organizations have created database of isolated words in many Indian languages. A multi-speaker, annotated and hand-segmented speech database for Hindi language was developed about a decade ago [4]. However, large spoken language corpora are yet to be generated for Indian languages.

3 Present Initiative

Development of Automatic Speech Recognition system in a language needs a large amount of annotated speech data in order to train statistical models as well as for evaluation of trained systems. Such corpora have to be created for many languages in a multi-lingual country such as India. An effort initiated by the spoken language group of TIFR is a step in this direction. The goal is to generate large corpora of spoken Indian languages. The corpora will consist of phonetically rich sentences spoken by many speakers as well application specific conversational speech. The database will be annotated at the lexical level.

Another feature of the proposed database is that speech will be collected over telephone channels. One of the unique features of voice oriented interface to computer is that there exists a telecommunication network that carries speech signal. This permits a user to conveniently access information from central databases remotely on anytime anywhere basis if an automatic speech recognition (ASR) and understanding system is part of the back-office. With the recent introduction of cellular networks, the telephone density in India has increased tremendously. Thus, it is necessary to collect narrow band speech data so that the database can be put to use to develop speech applications sooner.

In the first phase of the present programme, speech databases will be developed for four languages: Hindi, Marathi, Indian English and Malayalam. These languages have been chosen keeping in mind the importance, representation and ease of data acquisition. Hindi is the official language of the Government of India and spoken widely in North India. It is the mother-tongue of about one third of Indian population; another one third Indians speak Hindi as second language. TIFR is located in Mumbai, the capital of the western state of Maharashtra; Marathi is the official language of the state. English is an associate official language of the Government of India and is the *de facto* medium of instruction in colleges. Although its usage in India is much less than that of Hindi, it is spoken by educated Indians who happen to be the first users of modern technology. Also, because English is spoken as second or third language, the phrase structure of mother tongue influences that of spoken English. This gives rise to what can be called 'Indian English' whose lexicon and sentence formation differs noticeably from, say British or American English. In addition, the flavour of Indian English varies continuously across India due to a multitude of languages spoken.

Both Hindi and Marathi languages belong to the Indo-European language family, and in fact, share a common script. Even English belongs to the same language family. People in southern part of India speak languages belonging to Dravidian language family. Malayalam is a Dravidian language, and is the fourth language selected in the first stage of the effort.

This paper is a report of the preliminary work done in development of multilingual spoken language corpora development effort that was recently initiated at TIFR, Mumbai.

4 Phonetically Rich Text Corpora

Due to statistical nature of the models used by prevalent speech recognition systems, a large amount of speech data is needed to train the model. In order to represent (or at least to manage) variations in speech signal due to many factors such as phonetic context, speaker variability etc., complex models are used that involve a large number of parameters. For reliable estimation of these parameters, a lot of people have to speak many sentences that comprise of a variety of phonetic contexts. People are not willing to spend a lot of time reading dozens of sentences, since time is always at a premium. Moreover, data has to be collected from a large number of speakers. Thus, it is imperative to design compact sets of sentences that not only contain as many important phonetic contexts as possible, but also are not difficult to speak. A prerequisite for such a selection process is a large text corpus in a form that is amenable to such statistical analysis.

An easily available source of electronic text in Indian languages is the set of internet based vernacular newspapers. A few online newspapers have archives that go back 3 to 4 years. Another advantage is that this corpus keeps growing on a daily basis. However, there are some disadvantages too. A major disadvantage is that the text corpus gets dominated by the writing style of one newspaper. This problem can be alleviated if we can collect data from many newspapers. However, such an effort throws new challenges due to the nature of script used by Indian languages.

The Indic scripts used by Indian languages are syllabic in nature. Each character represents one vowel preceded by zero or more consonants. Figure 1 shows the basic characters of the Devanagari script used by Hindi and Marathi languages. The first panel shows vowels in independent form, i.e., no consonant. The second and third panels show consonants with an implicit /a/ vowel. The fourth panel illustrates syllabic characters comprising of phoneme /k/ following by various vowels; here, vowels are represented by glyphs called 'matras'. When

अ	आ	इ	ई	उ	ऊ	ए	ऐ	ओ	औ
a	aː	*i*	*i*ː	*u*	*u*ː	*e*	ae	*o*	au

क *k*	ख k^h	ग *g*	घ g^h	ङ ŋ
च tʃ	छ tʃ^h	ज ʤ	झ ʤ^h	ञ ɲ
ट ʈ	ठ ʈ^h	ड ɖ	ढ ɖ^h	ण ɳ
त *t*	थ t^h	द *d*	ध d^h	न *n*
प *p*	फ p^h	ब *b*	भ b^h	म *m*

य	र	ल	व	श	ष	स	ह
j	*r*	*l*	ʋ	ʃ	ʂ	*s*	*h*

क	का	कि	की	कु	कू	के	कै	को	कौ
ka	kA	ki	kI	ku	kU	ke	kE	ko	kO

Fig. 1. Basic characters of the Devanagari script used by Hindi and Marathi languages; the corresponding IPA symbols are also shown. The first panel shows vowels in independent form; the second and third panels show consonants with an implicit /a/ vowel. The fourth panel shows syllabic characters comprising of phoneme /k/ following by various vowels; vowels are represented by glyphs called 'matras'. Notice that quite a few graphemes share glyphs such as vertical lines. Also, a few characters are proper subsets of others. Exploitation of such properties to minimize the size of glyphs by font developers result in non-standard grapheme-to-phoneme conversion rules.

a syllable comprises of more than one consonant, consonants can be combined to form ligatures. This leads to hundreds of ligatures.

Notice that quite a few graphemes share glyphs such as vertical lines or semicircles. In fact, in case of graphemes with a vertical line (called danda) on right, the absence of the danda represents a pure consonant and is used to form ligatures. Also, a few characters are proper subsets of others. Such characteristics are exploited by certain font developers to construct a small set of glyphs that form the building blocks for constructing all characters including ligatures. For example, a consonant associated a simple grapheme could be encoded by one byte, whereas a consonant associated with a complex grapheme may be encoded by three bytes. Thus, in case of such fonts, the mapping between the glyphs and their binary codes is non-standard and needs to be discovered. On the other hand, if the publisher uses a standard font encoding scheme such as Unicode or UTF-8, it is easier to transliterate online vernacular text in roman script for gathering phonetic and linguistic statistics. Such a simple scheme was employed by a Hindi online newspaper. Thus it was easy to quickly acquire a large amount of text in electronic form and to select phonetically rich sentences as described in [6]. On the other hand, for Marathi language, we could not locate an online newspaper that used a well known encoding scheme. So, we had to discover the mapping, and write a complex program that derived the transliteration form. Same was found to be true with Malayalam language. For English language, we plan to use sentences of the TIMIT database [5].

Another disadvantage of using newspapers as source of text is that the web material belongs to a small set of topics and is not general enough. Moreover, sentences to be read should not be long or difficult to comprehend. These difficulties are more likely to be encountered with newspaper text. Therefore, there is a need to augment such sources with others such as online webpages and stories.

Once a large set of textual material is available, one can select phonetically rich sentences to form sets of sentences to be read by subjects. We use phonetically 'rich' criteria rather than the traditional phonetically 'balanced' criteria because we wanted to enrich the frequency count of rare phonemes in the database. In our previous effort [4], the goal was not only to enrich the phonetic diversity, but also richness of Broad Acoustic Class *pairs*. This was necessary because of the small size of the database. Also, the phoneme boundaries in speech data were to be manually marked; the database was supposed to serve other needs in addition to speech recognition. However, in the current programme, we restricted the goal to enhance the phonemic richness since the primary aim of the database is to train speech recognition models. The concern for diversity of phonetic context is likely to be less severe here due to large amount of speech data that is proposed to be collected. We plan to use the public domain software,"CorpusCrt", for this purpose [7]. It may be noted that phonetically rich sentences in the four languages designed in this project are not related to each other, unlike in the case of a parallel corpus.

5 Spoken Dialog

When human beings exchange information through natural language, the process does not complete with each person saying one sentence. Instead, a conversation takes place between human beings. This is quite often needed since people are wont to give incomplete or ambiguous information or query. The process of disambiguation or seeking clarification leads to a dialogue. The same phenomenon is likely to recur in case of voice oriented human machine interaction as well. So, developers of real-life speech applications would need to pay attention not only to speech recognition but also to other aspects of spoken dialogue. Such a dialogue is associated with spontaneous speech whose characteristics are quite different from read speech. Spontaneous speech contains speech disfluencies such as "ah"s, "am"s, pause, incomplete word, repetition, false starts etc.

Such natural human behaviour has to modeled in speech recognition systems. Any modeling needs data. Thus, collection of speech data in the context of a human machine conversation is necessary. In the absence of a machine with speech recognition/understanding capability, such a data can only be generated in a simulated environment. We plan to record speech of a person in 'conversation' with a machine with the goal of acquiring a specific information. The next section describes the data collection setup and the procedure for collecting read as well as spontaneous speech data.

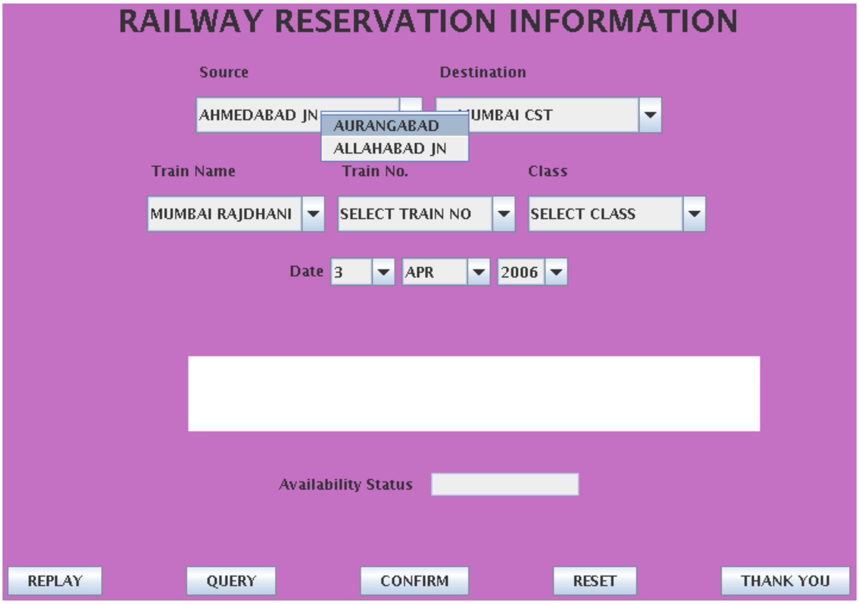
6 Data Acquisition Setup

The goal is to collect speech data over telephone channel; so, a computer telephony interface is a prerequisite. Since conversational speech is to be recorded, the interface must be intelligent enough to detect 'barge-in' by the subject and take appropriate action. A 4 port Dialogic analog voice card (Model D/41 JCT) was chosen to operate in Linux environment. The card can detect voice activity on the incoming channel, detect barge-in, and stop a system prompt playing on the outgoing channel of the same telephone line and record speech data in an echo canceled mode. It can store speech data either in a file or in an array that can be processed by another programme. This features enables automatic speech recognition in a live mode.

6.1 Wizard of Oz Setup

In addition to recording read speech, an important aim of the current speech database programme is to record goal oriented, conversational speech. Spontaneous speech has a few distinct characteristics in comparison to read speech. In the latter, speakers carefully read grammatically correct sentences designed to cover a variety of phonetic contexts and suprasegmental effects. On the other hand, spontaneous speech quite often includes ungrammatical phrases, and a variety of speech disfluencies as mentioned earlier. Therefore, acoustic and language models trained with read speech data is not suitable for conversational systems. Hence, it is necessary to collect speech data in a simulated environment where speakers feel that they are talking and listening to a machine. 'Wizard of Oz' is one such speech data collection method where a human volunteer acts like a wizard, and emulates the response of an ASR system. He always listens to the speech of the caller, but can never talk to the caller. Instead, he types in the likely response of the machine based on its understanding of the caller's speech. The machine takes the textual input entered by the wizard and generates corresponding speech; this synthesized speech is played to the caller. In addition, some errors of the ASR system are feigned so that the caller feels as if he is actually carrying on a spoken dialogue with the system. The acoustic and language models trained with speech data collected under such an audio-only environment are likely to perform better in actual usage by the general public.

The task domain of the current setup at TIFR is that of a railway reservation enquiry system. Figure 2 shows a snapshot of the java based Graphical User



RAILWAY RESERVATION INFORMATION

Source: AHMEDABAD JN

Destination: UMBAI CST

Train Name: MUMBAI RAJDHANI

Train No.: SELECT TRAIN NO

Class: SELECT CLASS

Date: 3 APR 2006

Availability Status

Fig. 2. A graphical user interface for Wizard of Oz experiment for collection of conversational speech over telephone channel under a simulated environment

Interface (GUI) that helps to simulate the response of a computer to the caller. The caller will be instructed to carry out a free form dialogue with the computer and achieve a pre-specified goal. A typical goal is to find out whether a reserved ticket of a certain class of travel is available for traveling between pre-specified stations by a train on a specified date. All interaction is in audio-only mode since the caller may be calling over mobile phone from anywhere.

A volunteer (the wizard) sits in front of the computer running the GUI and keeps listening to whatever the caller says. Mandatory information to be spoken by the caller are source and destination stations, train name or number, class and date of travel. Whenever the caller provides one or more such information, the wizard enters that information by selecting the suitable entry in a drop down menu in the appropriate panel. The caller is free to speak such that the utterance contains zero or one or many items of information. Moreover, he may provide information in any order. For instance, the destination and date of travel may be mentioned in the first query. When the wizard enters these information and presses the button “QUERY” at the lower part of the GUI, the system discovers that the source station is not known. It generates a query to this effect, synthesizes the corresponding speech waveform, and plays it over the telephone channel. The query also contains the information just provided by the caller so that he has the option of correcting the misunderstanding of the system, if any. The computer continues to carry on such a dialogue with the caller until all the necessary information is received for seeking the availability status from the database. Then, the wizard presses the “CONFIRM” button. The computer retrieves the information from the reservation information database, composes the response containing the availability status and plays the corresponding synthetic speech data. Then, it asks whether the user wants any further information. If the answer is affirmative, the wizard presses “RESET” button; the information on the panels get cleared. Else, the wizard chooses the “THANK YOU” button and the system plays a valedictory message.

It may be noted that the wizard never speaks and the caller hears only the synthesized speech. Also, the wizard has the option of deliberately entering incorrect information, thus simulating errors in speech recognition. Since these errors are normally among acoustically confusing words, the GUI facilitates simulation of such errors. Normally, selections in the windows are made by clicking the left button. On the other hand, when the wizard clicks the right button in station name or train name, a new pane appears containing phonetically similar items. Figure 2 illustrates such an event. The caller has spoken “Ahmedabad”; the system has shown that “Aurangabad” and “Allahabad” are city names that are acoustically similar to the spoken word ‘Ahmedabad’, and are potential outcome of ASR system in case of a misrecognition. The wizard can choose any of them in order to simulate an error of ASR system. In this fashion, response of callers to erroneous recognition of the machine can be collected and modeled. The speech and the corresponding text data can be used to train acoustic and language models for spoken conversation.

This setup is just to collect spontaneous speech data in a simulated environment. The system does not carry out input validation tests. For example, the station may not be on the itenery or path of the train. The train number, direction of travel, available classes may not match. The availability information provided at the end of the dialogue is purely random.

Currently, the java based GUI system and C based computer telephony interface software have been tested independently. The two are being integrated for the purpose of recording spoken conversations in Hindi. It can be easily adapted to other languages by changing the speech synthesis system. In the current version, we plan to use pre-recorded speech instead of a full blown text-to-speech system. We expect the data collection setup to be operational in about a month’s time.

We want the collected data to reflect the actual usage conditions. The only restriction we place currently is that the signal-to-noise ratio should not be low. It may be possible to relax this constraint in future by considering the fact that most mobile companies employ sophisticated signal processing algorithms to enhance speech of the speaker even when the microphone is several centimeters away and above the mouth. We plan to collect data from people with diverse educational, dialectal backgrounds. We also plan to collect data from many geographical locations. Also, we would like people to speak using their handset over landline or wireless channel. Such a diversity of handsets has to be strived for since in actual application, speech can come from any of the handsets. The wide variation in transduction properties of handsets is a major cause of misrecognition in ASR. In order detect the type of handset, we plan to collect information about the model of the handset. Additional information we would collect about speaker are gender, age group (3 groups: $<= 15, 15 < age <= 30, > 30$) and mother tongue.

One of the practical difficulties is to coax people to spend time to participate in the data collection process. Moreover, we expect subjects to use their phone (and pay for the call) because we want to capture variability in handset characteristics. So, we plan to entice people to cooperate with us by giving mobile recharge coupons that more than compensate for the phone bill. Still, volunteers are needed to coax people in India. The proposed strategy is to approach friends and relatives of volunteers first and approach others later hoping that the word of mouth spread of the news and incentive would do the needful. If a volunteer is present with the speaker, he can (a) give a hardcopy of the set of sentences to read, (b) explain the conversational data collection setup, and issue the card that contains the goal of that conversation to the subject, (c) and handover the coupons.

There is an additional advantage in having a volunteer with the subject during the recording process. While the subject is talking to the computer over the phone, the same speech can also be simultaneously recorded over a wideband speech recorder. The advent of small, flash drive based voice recorders has made this possible. The recorder is light and smaller than normal mobile phones in India, and thus can be attached to the mobile phone easily. The recorder has 1GB storage and has excellent speech I/O hardware. Thus, we plan to acquire broadband speech as well. This recording, however, will not have sentence boundary, may have the voice of volunteer and others, and will have to be processed by humans even to extract sentences. Yet, as the adage goes, “no data like more data”; one can even say “no data like real data”.

We have thoroughly tested the read speech data acquisition setup, by conducting trial runs in the lab, by making phone calls from suburbs and over mobile phones. We have standardized the speaker information sheet, file naming convention and speech data organization. We have also collected speech data from about a dozen persons. However, we would like to wait for the Wizard of Oz setup to be ready so that we can collect both type of data at one shot. We also plan to use the speech data collection drive to acquire handwritten Devanagari script data for online cursive script recognition studies.

6.2 Follow-Up Process

The previous section has described the data collection setup and the acquisition process in great detail. The recorded speech, however, has to be processed so that it can be used for developing ASR systems. While speech data can be automatically organized in speaker-specific directory structures, data still has to be validated for errors such as ‘no speech’ etc. Another time-consuming task is that of manual transcription of data by listening to spontaneous speech. The transcription involves not only noting down the sequence of words but also speech disfluencies and pauses. A pronunciation dictionary has to be prepared that contains all the words that occur in the conversation. As a reviewer pointed out, it is desirable to validate the recorded data and transcription process at an early stage. An evaluation of the initial data would give an opportunity to improve and modify the data collection process, if it is necessary. In view of the task-specific, goal oriented conversation, a study of various dialogue samples may give us clues to generate language models (for use in automatic speech recognition) that are better than a simple n-gram language model.

7 Conclusions

Generation of multi-lingual speech corpora for Indian languages is a dire necessity if the benefit of information technology has to be made available to all citizens of the country. We have presented an outline of a recent initiative to collect speech data in 4 Indian languages and described the preparatory work that has been completed with respect to Hindi. Read speech data has been collected from about a dozen speakers of Hindi language, and trial runs for collection of spontaneous speech data will commence soon. Preparatory work for extending the data collection process to other 3 Indian languages is in progress. We hope to collect significant amount of speech data that will bring us closer to face challenges as well as to realize opportunities in the development of spoken language machine interfaces for use by ordinary Indians.

References

1. http://au-kbc.org/dfki/index.html
2. http://tdil.mit.gov.in/corpora/ach-corpora.htm
3. S. Agrawal, K. Samudravijaya and Karunesh Arora: Recent Advances of Speech Databases development activity for Indian Languages. Proc. of ISCSLP 2006, Companion Volume, published by COLIPS, Singapore, December, 2006
4. Samudravijaya K, Rao, P.V.S., Agrawal S.S.: Hindi Speech Database. Proc. Int. Conf. on Spoken Language processing(ICSLP00) Beijing China, 2000, CDROM paper: 00192.pdf
5. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S1
6. Chourasia V., Samudravijaya K., Chandwani M.: Phonetically Rich Hindi Sentence Corpus for Creation of Speech Database. Proc. O-COCOSDA 2005, Indonesia, pp. 132–137.
7. http://gps.tsc.upc.es/veu/personal/sesma/sesma/CorpusCrt/php3

Author Index