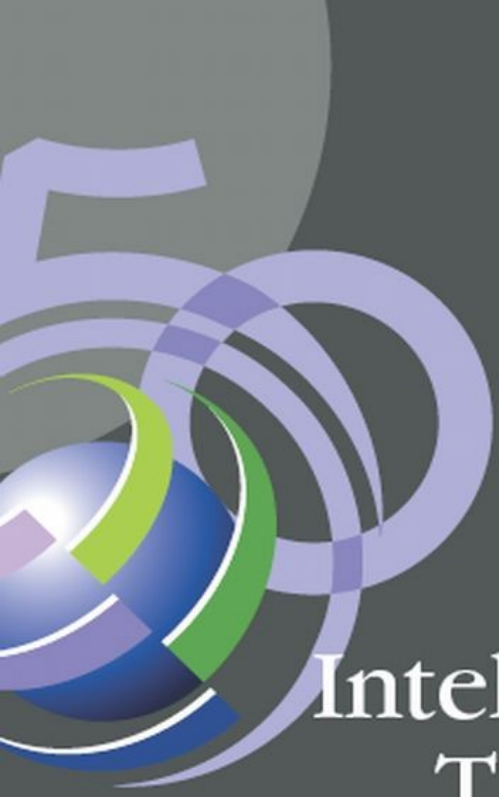


Artificial Intelligence in Theory and Practice II

WCC 2008 Milano, Italy

Edited by
Max Bramer

ARTIFICIAL INTELLIGENCE IN THEORY AND PRACTICE II

IFIP – The International Federation for Information Processing

IFIP was founded in 1960 under the auspices of UNESCO, following the First World Computer Congress held in Paris the previous year. An umbrella organization for societies working in information processing, IFIP's aim is two-fold: to support information processing within its member countries and to encourage technology transfer to developing nations. As its mission statement clearly states,

> *IFIP's mission is to be the leading, truly international, apolitical organization which encourages and assists in the development, exploitation and application of information technology for the benefit of all people.*

IFIP is a non-profitmaking organization, run almost solely by 2500 volunteers. It operates through a number of technical committees, which organize events and publications. IFIP's events range from an international congress to local seminars, but the most important are:

- The IFIP World Computer Congress, held every second year;
- Open conferences;
- Working conferences.

The flagship event is the IFIP World Computer Congress, at which both invited and contributed papers are presented. Contributed papers are rigorously refereed and the rejection rate is high.

As with the Congress, participation in the open conferences is open to all and papers may be invited or submitted. Again, submitted papers are stringently refereed.

The working conferences are structured differently. They are usually run by a working group and attendance is small and by invitation only. Their purpose is to create an atmosphere conducive to innovation and development. Refereeing is less rigorous and papers are subjected to extensive group discussion.

Publications arising from IFIP events vary. The papers presented at the IFIP World Computer Congress and at open conferences are published as conference proceedings, while the results of the working conferences are often published as collections of selected and edited papers.

Any national society whose primary activity is in information may apply to become a full member of IFIP, although full membership is restricted to one society per country. Full members are entitled to vote at the annual General Assembly, National societies preferring a less committed involvement may apply for associate or corresponding membership. Associate members enjoy the same benefits as full members, but without voting rights. Corresponding members are not represented in IFIP bodies. Affiliated membership is open to non-national societies, and individual and honorary membership schemes are also offered.

ARTIFICIAL INTELLIGENCE IN THEORY AND PRACTICE II

IFIP 20th World Computer Congress, TC 12: IFIP AI 2008 Stream, September 7-10, 2008, Milano, Italy

Edited by

Max Bramer
University of Portsmouth
United Kingdom

Springer

Library of Congress Control Number: 2008929520

Artificial Intelligence in Theory and Practice II

Edited by Max Bramer

p. cm. (IFIP International Federation for Information Processing, a Springer Series in Computer Science)

ISSN: 1571-5736 / 1861-2288 (Internet)
ISBN: 978-0-387-09694-0
eISBN: 978-0-387-09695-7

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

IFIP 2008 World Computer Congress (WCC'08)

Message from the Chairs

Every two years, the International Federation for Information Processing hosts a major event which showcases the scientific endeavours of its over one hundred Technical Committees and Working Groups. 2008 sees the 20th World Computer Congress (WCC 2008) take place for the first time in Italy, in Milan from 7-10 September 2008, at the MIC - Milano Convention Centre. The Congress is hosted by the Italian Computer Society, AICA, under the chairmanship of Giulio Occhini.

The Congress runs as a federation of co-located conferences offered by the different IFIP bodies, under the chairmanship of the scientific chair, Judith Bishop. For this Congress, we have a larger than usual number of thirteen conferences, ranging from Theoretical Computer Science, to Open Source Systems, to Entertainment Computing. Some of these are established conferences that run each year and some represent new, breaking areas of computing. Each conference had a call for papers, an International Programme Committee of experts and a thorough peer reviewed process. The Congress received 661 papers for the thirteen conferences, and selected 375 from those representing an acceptance rate of 56% (averaged over all conferences).

An innovative feature of WCC 2008 is the setting aside of two hours each day for cross-sessions relating to the integration of business and research, featuring the use of IT in Italian industry, sport, fashion and so on. This part is organized by Ivo De Lotto. The Congress will be opened by representatives from government bodies and Societies associated with IT in Italy.

This volume is one of fourteen volumes associated with the scientific conferences and the industry sessions. Each covers a specific topic and separately or together they form a valuable record of the state of computing research in the world in 2008. Each volume was prepared for publication in the Springer IFIP Series by the conference's volume editors. The overall Chair for all the volumes published for the Congress is John Impagliazzo.

For full details on the Congress, refer to the webpage http://www.wcc2008.org.

Judith Bishop, South Africa, Co-Chair, International Program Committee
Ivo De Lotto, Italy, Co-Chair, International Program Committee
Giulio Occhini, Italy, Chair, Organizing Committee
John Impagliazzo, United States, Publications Chair

WCC 2008 Scientific Conferences

TC12	**AI**	Artificial Intelligence 2008
TC10	**BICC**	Biologically Inspired Cooperative Computing
WG 5.4	**CAI**	Computer-Aided Innovation (Topical Session)
WG 10.2	**DIPES**	Distributed and Parallel Embedded Systems
TC14	**ECS**	Entertainment Computing Symposium
TC3	**ED_L2L**	Learning to Live in the Knowledge Society
WG 9.7	**HCE3**	History of Computing and Education 3
TC3		
TC13	**HCI**	Human Computer Interaction
TC8	**ISREP**	Information Systems Research, Education and Practice
WG 12.6	**KMIA**	Knowledge Management in Action
TC2	**OSS**	Open Source Systems
WG 2.13		
TC11	**IFIP SEC**	Information Security Conference
TC1	**TCS**	Theoretical Computer Science

IFIP

- is the leading multinational, apolitical organization in Information and Communications Technologies and Sciences
- is recognized by United Nations and other world bodies
- represents IT Societies from 56 countries or regions, covering all 5 continents with a total membership of over half a million
- links more than 3500 scientists from Academia and Industry, organized in more than 101 Working Groups reporting to 13 Technical Committees
- sponsors 100 conferences yearly providing unparalleled coverage from theoretical informatics to the relationship between informatics and society including hardware and software technologies, and networked information systems

Details of the IFIP Technical Committees and Working Groups can be found on the website at http://www.ifip.org.

Contents

Foundations

Applications 1

Applications 2

Foreword

The papers in this volume comprise the refereed proceedings of the conference 'Artificial Intelligence in Theory and Practice' (IFIP AI 2008), which formed part of the 20th World Computer Congress of IFIP, the International Federation for Information Processing (WCC-2008), in Milan, Italy in September 2008.

The conference is organised by the IFIP Technical Committee on Artificial Intelligence (Technical Committee 12) and its Working Group 12.5 (Artificial Intelligence Applications).

All papers were reviewed by at least two members of our Program Committee. Final decisions were made by the Executive Program Committee, which comprised John Debenham (University of Technology, Sydney, Australia), Ilias Maglogiannis (University of Aegean, Samos, Greece), Eunika Mercier-Laurent (KIM, France) and myself. The best papers were selected for the conference, either as long papers (maximum 10 pages) or as short papers (maximum 5 pages) and are included in this volume. The international nature of IFIP is amply reflected in the large number of countries represented here.

The conference also featured invited talks by Prof. Nikola Kasabov (Auckland University of Technology, New Zealand) and Prof. Lorenza Saitta (University of Piemonte Orientale, Italy).

I should like to thank the conference chair, John Debenham for all his efforts and the members of our program committee for reviewing papers to a very tight deadline.

This is the latest in a series of conferences organised by IFIP Technical Committee 12 dedicated to the techniques of Artificial Intelligence and their real-world applications. The wide range and importance of these applications is clearly indicated by the papers in this volume. Further information about TC12 can be found on our website http://www.ifiptc12.org.

Max Bramer
Chair, IFIP Technical Committee on Artificial Intelligence

Acknowledgements

Conference Organising Committee

Conference General Chair
John Debenham (University of Technology, Sydney, Australia)

Conference Program Chair Max Bramer (University of Portsmouth, United Kingdom)

Executive Program Committee

Max Bramer (University of Portsmouth, United Kingdom)
John Debenham (University of Technology, Sydney, Australia)
Ilias Maglogiannis (University of Aegean, Samos, Greece)
Eunika Mercier-Laurent (KIM, France)

Program Committee

Analia Amandi (ISISTAN Research Institute, Argentina)
Fabrizio Angiulli (DEIS, Università della Calabria, Italy)
Stefania Bandini (University of Milan, Italy)
Cristian Barrué (Technical University of Catalonia, Barcelona, Spain)
Daniel Berrar (University of Ulster, Northern Ireland)
Alina Bogan-Marta (University of Oradea, Romania)
Max Bramer (University of Portsmouth, United Kingdom)
Maricela Bravo (Universidad Politécnica del Estado de Morelos)
Per Bjarne Bro (Chile)
Krysia Broda (Imperial College, London, United Kingdom)
Luigia Carlucci Aiello (Università di Roma La Sapienza, Italy)
Ana Casali (FCEIA - UNR, Argentina)
Fabrice Colas (Leiden University, The Netherlands)
John Debenham (University of Technology, Sydney, Australia)
Yves Demazeau (CNRS - LIG Grenoble, France)
Vladan Devedzic (University of Belgrade, Serbia)
Tharam Dillon (Curtin University of Technology, Australia)

Graçaliz Pereira Dimuro (Universidade Católica de Pelotas, Brazil)
Anne Dourgnon-Hanoune (EDF, France)
Gintautas Dzemyda (Institute of Mathematics and Informatics, Lithuania)
Henrik Eriksson (Linköping University, Sweden)
Enrique Ferreira (Universidad Católica del Uruguay)
Anabel Fraga (Spain)
Matjaz Gams (Slovenia)
Ramon Garcia-Martinez (Buenos Aires Institute of Technology, Argentina)
Ana Garcia-Serrano (Technical University of Madrid, Spain)
Martin Josef Geiger (University of Hohenheim, Germany)
João Gluz (Brazil)
Daniela Godoy (ISISTAN Research Institute, Argentina)
Enrique González (Pontificia Universidad Javeriana, Colombia)
Andreas Harrer (Catholic University Eichstätt-Ingolstadt, Germany)
Kostas Karpouzis (National Technical University of Athens, Greece)
Nik Kasabov (Auckland University of Technology, New Zealand)
Dusko Katic (Serbia and Montenegro)
Nicolas Kemper Valverde (Universidad Nacional Autónoma de México)
Joost N. Kok (Leiden University, The Netherlands)
Piet Kommers (University of Twente, The Netherlands)
Jasna Kuljis (Brunel University, United Kingdom)
Daoliang Li (China Agricultural University, Beijing)
Aurelio Lopez-Lopez (Instituto Nacional de Astrofisica, Optica y Electronica,
Mexico)
Ramon Lopez de Mantaras (Spanish Council for Scientific Research)
Ilias Maglogiannis (University of Aegean, Samos, Greece)
Suresh Manandhar (University of York, United Kingdom)
Edson Takashi Matsubara (University of Sao Paulo (USP) Brazil)
Brian Mayoh (University of Aarhus, Denmark)
Eunika Mercier-Laurent (KIM, France)
Tanja Mitrovic (University of Canterbury, Christchurch, New Zealand)
Riichiro Mizoguchi (Osaka University, Japan)

Pavol Navrat (Slovak University of Technology in Bratislava, Slovakia)
Adolfo Neto (Federal Center for Technological Education- CEFET-SP, Brazil)
Erich J. Neuhold (University of Vienna)
Bernd Neumann (University of Hamburg, Germany)
Toyoaki Nishida (Kyoto University)
Daniel O'Leary (University of Southern California, USA)
Andrea Omicini (Alma Mater Studiorum - Università di Bologna, Italy)
Mihaela Oprea (University of Ploiesti, Romania)
Alun Preece (University of Aberdeen, United Kingdom)
Joel Prieto Corvalán (Universidad Católica de Asunción, Paraguay)
Cristian Rusu (Pontificia Universidad Católica de Valparaíso, Chile)
Abdel-Badeeh M. Salem (Ain Shams University, Egypt)
Demetrios Sampson (University of Piraeus & CERTH, Greece)
Silvia Schiaffino (ISISTAN Research Institute, Argentina)
Valery Sklyarov (Portugal)
Mauricio Solar (University of Santiago de Chile)
Constantine Spyropoulos (Inst. of Informatics & Telecommunications, Greece)
Georgina Stegmayer (CIDISI Research Institute-CONICET, Argentina)
Olga Stepankova (Czech Technical University in Prague, Czech Republic)
Péter Szeredi (Budapest University of Technology and Economics, Hungary)
Ahmed Tawfik (University of Windsor)
Eric Tsui (The Hong Kong Polytechnic University)
Manuel Tupia (Ponticia Universidad Católica del Perú)
Wiebe van der Hoek (University of Liverpool, United Kingdom)
Marcos Villagra (Catholic University of Asuncion, Paraguay)
Richard Weber (University of Chile, Chile)
Jin-Mao Wei (Nankai University, China)
Ting Yu (University of Sydney, Australia)
Zdenek Zdrahal (The Open University, United Kingdom)
Alejandro Zunino (Argentina)

AGENTS 1

A Light-Weight Multi-Agent System Manages 802.11 Mesh Networks

Ante Prodan and John Debenham

Abstract A light-weight multi-agent system is employed in a "self-organisation of multi-radio mesh networks" project to manage 802.11 mesh networks. As 802.11 mesh networks can be extremely large the two main challenges are the scalability and stability of the solution. The basic approach is that of a distributed, light-weight, co-operative multiagent system that guarantees scalability. As the solution is distributed it is unsuitable to achieve any global optimisation goal — in any case, we argue that *global optimisation* of mesh network performance in any significant sense is not feasible in real situations that are subjected to unanticipated perturbations and external intervention. Our overall goal is simply to reduce maintenance costs for such networks by removing the need for humans to tune the network settings. So stability of the algorithms is our main concern.

1 Introduction

The work discussed is based on previous work in the area of mesh networking and in particular in distributed algorithms at Columbia University, Microsoft Research, University of Maryland and Georgia Institute of Technology. In particular: [1], [2], [3] and [4].

Recent work on 802.11 Mesh Networks, such as [5], is predicated on a network whose prime purpose is to route traffic to and from nodes connected to the wired network — in which case there is assumed to be no traffic between end-user nodes. This introduces the conceptual simplification that mesh nodes can be seen as being grouped into clusters around a wired node where each cluster has a tree-like struc-

Ante Prodan
University of Technology, Sydney, Australia e-mail: aprodan@it.uts.edu.au

John Debenham
University of Technology, Sydney, Australia e-mail: debenham@it.uts.edu.au

Please use the following format when citing this chapter:

Prodan, A. and Debenham, J., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 3–12.

ture, rooted at a wired node, that supports the traffic. This is the prime purpose of 802.11 Mesh Networks in practice. In the work that follow we have, where possible, moved away from any assumptions concerning tree-like structures with the aim of designing algorithms for quite general mesh networks. Our methods have, where possible, been designed for the more general classes of "wireless ad-hoc networks" or "wireless mesh networks".

There are three principal inputs to this work that we assume are available to the proposed methods:

- A load model. Given any contiguous set of nodes in a mesh, the *load model* specifies the actual or desired level of traffic flowing into, or out of, nodes in that set.
- A load balancing algorithm. Given any contiguous set of nodes in a mesh and the load model for that set, the *load balancing algorithm* determines how the traffic is allocated to links in the mesh so as to reach its desired destination where it leaves the mesh.
- An interference model. Given any contiguous set of nodes in a mesh, the *interference model* stipulates the interference level that each node in the mesh gives to the other nodes in the mesh given a known level of background interference due to transmission devices that are external to the mesh.

The work described below makes no restrictions on these three inputs other than that they are available to every node in the mesh. The load model, and so too the load balancing algorithm, will only be of value to a method for self-organisation if together they enable future load to be predicted with some certainty. We assume that the load is predictable.

In Section 2 we introduce some terms, concepts and notation. Section 3 describes the illocutions that make up the communication language used by the light-weight co-operative multiagent system that achieves self-organisation. We describe the role of the load balancing algorithm that our methods take as a given input. The measurement of interference cost is discussed in Section 4. Methods for the adjusting the channels in a multi-radio mesh networks for predictable load are described in Section 5, as well as a method for adjusting the links. Future plans are described in Section 6.

2 Basic terms and concepts

The discrete time intervals mentioned below, e.g. t, $t+1$, are sufficiently spaced to permit what has to be done to be done.

Available channels: $1,\ldots,K$.

A *node* is a set of radio interfaces (or "antennae") where each *interface* is associated with a particular *channel*, together with a controller that (intelligently we hope) assigns the channel on each interface.

A *link* is a pair of interfaces where each interface is assigned the same channel. The idea is that two interfaces communicate through a shared link. That is, if an interface is part of a link its state will be "listening and transmitting", otherwise its state will be "listening only".

Notation: nodes are denoted by Latin letters: a, b, c,..., the interfaces for node a are denoted by: $a[i]$ for $i = 1,\ldots$, and links are denoted by Greek letters: α, β, γ,.... The interfaces communicate using an illocutionary communication language that is defined informally (for the time being) with illocutions being encapsulated in quotation marks: "·".

For any node n, S_n is the set of nodes in node n's interference range. Likewise, for any link α, S_α is the set of links that contain nodes n's interference range $\forall n \in \alpha$.

Given a node a, define $V_a = \cup_{n \in S_a} S_n$.

Γ_x^t is channel used by x to communicate at time t where x may be either an interface or a link.

$f(\cdot,\cdot)$ is an *interference cost function* that is defined between two interfaces or two links. It estimates the cost of interference to one interface caused by transmission from the other interface. This function relies on estimates of the interference level and the level of load (i.e.: traffic volume). So this function requires an *interference model* and a *load model*. This function is described in Section 4.

An interface is either 'locked' or 'unlocked'. A locked interface is either locked because it has committed to lock itself for a period of time on request from another interface, or it is 'self-locked' because it has recently instigated one of the self-organisation procedures in Section 5. A locked interface is only locked for a 'very short' period during the operation of each of those procedures. This is simply to ensure that no more than one alteration is made during any one period — this is necessary to ensure the stability of the procedures. We also say that a node is locked meaning that all the interfaces at that node are locked.

The abbreviation SNIR means "signal to noise plus interference ratio".

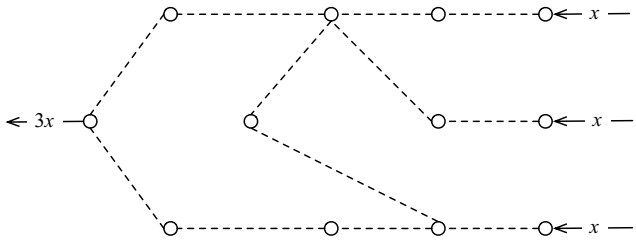
802.11 related terms: BSS — the basic service set. Portal — is the logical point at which MSDUs from an integrated non-IEEE 802.11 LAN enter the IEEE 802.11 DS (distribution system). WM — Wireless Medium. IBSS — Independent Basic Service Set. MSDU — MAC Service Data Unit.

3 The Communication Language

Multiagent systems communicate in illocutionary languages. The simple language defined here will in practice be encoded as a small block in a packet's payload.

- "**propose organise**$[a,b,p]$" sent from interface a to interface $b \in V_a$, where V_a is as above. This message advises interface b that interface a intends to instigate the proactive logic with priority p.
- "**overrule organise**$[a,b,q]$" sent from interface b to interface a. This message advises interface a that interface b intends to issue a **propose organise** statement

Fig. 1 The load balancing algorithm determines the allocation of load.



as it has priority $q > p$. That is an interface can only overrule a request to organise if it has higher priority.

The following three illocutions refer to interfaces being "locked" — this is simply a device to prevent interfaces from adjusting their settings when interference measurements are being made.

- "**propose lock**$[a,b,s,t]$" sent from interface a to interface b requests that interface b enter the locked state for the period of time $[s,t]$.
- "**accept lock**$[a,b,s,t]$" sent from interface b to interface a commits to interface b entering the locked state for the period of time $[s,t]$.
- "**reject lock**$[a,b,s,t]$" sent from interface b to interface a informs interface a that interface b does not commit entering the locked state for the period of time $[s,t]$.

4 Measuring Interference Cost

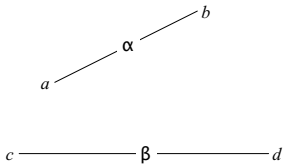
Suppose that during some time interval Δt two interfaces a and b are transmitting and receiving on channels Γ_a and Γ_b. During Δt, the *interference limit* that interface x imposes on interface y, $\tau_{y|x}$, is a ratio being the loss of traffic volume that interface y could receive if interface x were to transmit persistently divided by the volume of traffic that interface y could receive if interface x was silent:

$$\tau_{y|x} = \frac{(m_y \mid \text{ interface } x \text{ silent}) - (m_y \mid \text{ interface } x \text{ persistent})}{m_y \mid \text{ interface } x \text{ silent}}$$

where m_y is the mean SNIR observed by interface y whilst listening on channel Γ_y, where as many measurements are made as is expedient in the calculation of this mean[1]. The *interference load* of each interface, v_a and v_b, is measured as a proportion, or percentage, of some time interval during which that interface is transmitting. Then the *observed interference* caused by interface b transmitting on channel Γ_b as

[1] For $\tau_{y|x}$ to have the desired meaning, m_y should be a measurement of *link throughput*. However, link throughput and SNIR are approximately proportional — see [6].

Fig. 2 Definition of $f(\alpha \mid \beta)$.



experienced by interface a listening on channel Γ_a is: $\tau_{a|b} \times v_b$, and the *observed interference cost* to interface a is[2]:

$$f(a \mid b) \triangleq \tau_{a|b} \times v_b \times (1 - v_a)$$

and so to interface b:

$$f(b \mid a) = \tau_{b|a} \times v_a \times (1 - v_b)$$

Now consider the interference between one interface a and two other interfaces c and d. Following the argument above, the *observed interference* caused by interfaces c and d as experienced by interface a is[3]: $\tau_{a|c} \times v_c + \tau_{a|d} \times v_d - \tau_{a|\{c,d\}} \times v_c \times v_d$. The observed interference cost to interface a is:

$$f(a \mid \{c,d\}) = (1 - v_a) \times \left(\tau_{a|c} \times v_c + \tau_{a|d} \times v_d - \tau_{a|\{c,d\}} \times v_c \times v_d\right)$$

If interfaces c and d are linked, as shown in Figure 2, then they will transmit on the same channel Γ_β, and we ignore the possibility of them both transmitting at the same time[4]. Further suppose that v_β is the proportion of Δt for which either interface c or interface d is transmitting. Then for some κ_β, $0 \leq \kappa_\beta \leq 1$: $v_c = \kappa_\beta \times v_\beta$, and $v_d = (1 - \kappa_\beta) \times v_\beta$. Thus:

$$f(a \mid \beta) = (1 - v_a) \times v_\beta \times \left(\tau_{a|c} \times \kappa_\beta + \tau_{a|d} \times (1 - \kappa_\beta)\right)$$

Now suppose that interfaces a and b are linked, and that v_α is the proportion of Δt for which either interface a or interface b is transmitting. Then for some κ_α, $0 \leq \kappa_\alpha \leq 1$: $v_a = \kappa_\alpha \times v_\alpha$, $v_b = (1 - \kappa_\alpha) \times v_\alpha$. Then as a will only receive interference when it is listening to b transmitting:

$$f(a \mid \beta) = v_b \times v_\beta \times \left(\tau_{a|c} \times \kappa_\beta + \tau_{a|d} \times (1 - \kappa_\beta)\right)$$

---

[2] We assume here that whether or not interfaces a and b are transmitting are independent random events [7]. Then the probability that a is transmitting at any moment is v_a, and the probability that b is transmitting and a is listening at any moment is: $(1 - v_a) \times v_b$.

[3] That is, the interference caused by either interface c or interface d.

[4] The probability of two linked interfaces transmitting at the some time on an 802.11 mesh network can be as high as 7% — see [8], [9].

and so:

$$
\begin{aligned}
f(\alpha \mid \beta) = & \\
& (1-\kappa_\alpha) \times v_\alpha \times v_\beta \times \left(\tau_{a|c} \times \kappa_\beta + \tau_{a|d} \times (1-\kappa_\beta)\right) \\
& + \kappa_\alpha \times v_\alpha \times v_\beta \times \left(\tau_{b|c} \times \kappa_\beta + \tau_{b|d} \times (1-\kappa_\beta)\right)
\end{aligned} \tag{1}
$$

Note that v_α, v_β, κ_α and κ_β are provided by the load model, and the $\tau_{x|y}$ are provided by the interference model.

5 Adjusting the channels

Our solution is based on the distinction in multiagent systems between proactive and reactive reasoning. Proactive reasoning is concerned with planning to reach some goal. Reactive reasoning is concerned with dealing with unexpected changes in the agent's environment. So in the context of self-organising networks we distinguish between:

- a *reactive logic* that deals with problems as they occur. The aim of our reactive module is simply to restore communication to a workable level that may be substantially sub-optimal.
- a *proactive logic* that, when sections of the network are temporarily stable, attempts to adjust the settings on the network to improve performance.

The reactive logic provides an "immediate fix" to serious problems. The proactive logic, that involves deliberation and co-operation of nearby nodes, is a much slower process.

A node (i.e.: router) with omnidirectional interfaces has three parameters to set for each interface: [1] The channel that is assigned to that interface; [2] The interfaces that that interface is linked to, and [3] The power level of the interface's transmission. Methods are describe for these parameters in the following sections. The following section describes how these three methods used combined in the proactive logic algorithm. The following methods all assume that there is a load balancing algorithm and that it is common knowledge. The following methods are independent of the operation of the load balancing algorithm.

Informally the proactive logic uses the following procedure:

- *Elect* a node a that will manage the process
- *Choose* a link α from a to another node — precisely a trigger criterion (see below) permits node a to attempt to improve the performance of one of its links $\alpha \ni a$ with a certain priority level.
- *Measure* the interference
- *Change* the channel setting if appropriate

The following is a development of the ideas in [1].

choose node a **at** time $t-2$;
set $V_a = \cup_{n \in S_a} S_n$;
$\forall x \in V_a$ **transmit** "**propose organise**$[a,x,p]$";
unless $\exists x \in V_a$ **receive** "**overrule organise**$[a,x,q]$" in
$[t-2,t-1]$ where $q > p$ **do** {
$\forall x \in V_a$ **transmit** "**propose lock**$[a,x,t,t+1]$";
if $\forall x \in V_a$ **receive** "**accept lock**$[a,x,t,t+1]$" in $[t-1,t]$
then {
unless $\exists x \in V_a$ **receive** "**reject lock**$[a,x,t,t+1]$"
do {improve a;}
}
}
where: improve $a = \{$
choose link $\alpha \ni a$ on channel Γ_α^t;
set $B \leftarrow \sum_{\beta \in S_\alpha} f(\alpha \mid \beta) + \sum_{\beta \in S_\alpha} f(\beta \mid \alpha)$;
if (feasible) **re-route** α's traffic;
for $\Gamma_\alpha = 1, \ldots, K, \Gamma_\alpha \neq \Gamma_\alpha^t$ **do**{
if $\sum_{\beta \in S_\alpha} f(\alpha \mid \beta) + \sum_{\beta \in S_\alpha} f(\beta \mid \alpha) < B \times \varepsilon$ **then**{
$\Gamma_\alpha^{t+1} \leftarrow \Gamma_\alpha$;
selflock node a **in** $[t+1,t+k]$;
break;
};
};
$\forall x \in V_a$ **transmit** "α's interference test signals";
apply load balancing algorithm to S_a;
}

The statement **selflock** is to prevent a from having to activate the method too frequently. The constant $\varepsilon < 1$ requires that the improvement be 'significant' both for node a and for the set of nodes S_a. The stability of this procedure follows from the fact that it produces a net improvement of the interference cost within S_a. If a change of channel is effected then there will be no resulting change in interference outside S_a.

The above method reduces the net observed inference cost in the region V_a. It does so using values for the variables that appear on the right-hand side of Equation 1. If those values are fixed then the method will converge. The method above suggests the possibility that traffic is re-routed during the reassignment calculation — this is not essential.

5.1 Interference model

We assume that each node, a, knows the channel of every node in V_a. We assume that each node is capable of measuring the strength of signals from every node in V_a. So if each node had access to all of this information from the point of view of every

node in V_a, and, perhaps the level of background noise around V_a then a can derive estimates for the $\tau_{x|y}$ factors for all x and y in V_a. In particular, a will be able to estimate all these factors to evaluate Equation 1 as required by the above algorithm. *In addition*, the procedure above suggests that if node a is involved in changing its channel then at the end of this process — time permitting — it should transmit a 'beep-silence-beep-silence' message to enable every other node in V_a to observe the actual τ values. *Further*, it is reasonable to suggest that this transmission of test signals could be carried out periodically in any case when network load permits.

5.1.1 Expected SNIR

The complete SNIR at the receiver based on a set of interfering links in the carrier sensing range is given by[5]

$$\begin{aligned} \text{SNIR} &= \frac{P_r}{N + \sum_{k=1}^{n} I_k} \\ N &= K \times W \times T \end{aligned} \tag{2}$$

where: P_r = Received power of the frame, $\sum_k I_k$ = Received powers of the set of n interfering nodes (interfaces), N = Thermal noise, k = Boltzmann constant, W = Spectral bandwidth of the carrier (For example the channel bandwidth is 22 MHz in 802.11b), and T = Absolute temperature at the receiver.

Let us assume that the node (interface) j wants to trigger the proactive logic i.e. possibly change channel with node (interface) i. Then Equation 2 gives the sum of the interferences from the neighbouring links[6] in the carrier sensing range:

$$\sum_{k=1}^{n} I_k = \sum_{<k,l>\in R} \frac{P_{kl} \times G_{jk} \times G_{kj}}{\text{PL}_{kj}}$$

where: R = Set of all links that interfere with link α between i and j node (interfaces), P_{kl} = Power transmitted by the node (interface) k to node (interface) l, G_{jk} = Gain of Antenna of node (interface) j towards node (interface) k, G_{kj} = Gain of Antenna of node (interface) k towards node (interface) j, and PL_{kj} = Path loss suffered by the signal while traversing from the node (interface) k to the node (interface) j.

The values for 802.11 interfaces transmit power, Antenna gains are generally specified by the vendor in the data sheets of the equipment.

A general formula for calculating path loss (PL) in the Friis free space i.e. Line of Sight (LOS) link between the transmitter and receiver is given by[7]

---

[5] Analyses of Measurements and Simulations in Multi-hop Ad-hoc Environment, Report IST-2001-37385 6HOP D2.3

[6] see "Topology Planning for Long Distance Wireless Mesh Networks", Indian Institute of technology, Kanpur.

[7] Simon Haykin, Communication Systems, 4th edition, John Wiley & Sons Inc, 2001.

$$\mathrm{PL} = -10 \times \log_{10}(G_t G_r) + 10 \times \log_{10}\left(\frac{4 \times \pi \times d}{\lambda}\right)^2$$

where: G_t = Gain of the transmitting antenna, G_r = Gain of the receiving antenna, d = Distance between the transmitting and receiving antennas, and λ= Transmission Wavelength. In our Wireless Mesh Network the GPS in the nodes can measure d.

However, in most of the scenarios for urban areas the link between the transmitter and receiver will generally be Non LOS (NLOS). In these cases we can determine the path loss PL_{kj} by using the Co-operation in the field of Scientific and Technical research project 231 (COST231) adopted propagation model called as the *Walfisch-Ikegami model*[8].

Therefore the formula for the expected SNIR is given by:

$$\mathbb{E}(\mathrm{SNIR}) = \frac{P_{ij} \times G_{ij} \times G_{ji}}{N + \sum_{<k,l>\in R} \mathrm{PL}_{kj}}$$

5.1.2 Expected BER and FER

The BER is based on the type of modulation scheme that is used by the PHY layer of the radio to transmit the data. For example 802.11b uses different modulation schemes for different data rates such as: Differential Binary Phase Shift Keying (DBPSK) for 1 Mbps, Differential Quadrature Phase Shift Keying (DQPSK) for 2 Mbps and Complimentary Code Keying (CCK) for 5.5 and 11 Mbps [9].

Each of the modulation schemes has a different formula for calculating the BER, which can be referred to in[10].

For example the BER in an Additive White Gaussian Noise (AWGN) channel for DBPSK is given by[11]:

$$\mathrm{BER} = \frac{1}{2} \times \exp(-\mathrm{SNIR})$$

Assuming that each bit error is an independent event, then a simple relationship between BER and FER is given by[12]:

$$\mathrm{FER} = 1 - (1 - \mathrm{BER})^n$$

where: n = Number of bits in the frame.

---

[8] J.S. Lee and L.E. Miller, CDMA Systems Engineering Handbook, Artech House, 1998.

[9] Ji Zhang, "Cross-Layer Analysis and Improvement for Mobility Performance in IP-based Wireless Networks", Ph.D. Thesis, Sept. 2005.

[10] A. Ranjan, "MAC Issues in 4G", IEEE ICPWC, pp. 487-490, 2005.

[11] Simon Haykin, Communication Systems, 4th edition, John Wiley & Sons Inc, 2001.

[12] Analyses of Measurements and Simulations in Multi-hop Ad-hoc Environment, Report IST-2001-37385 6HOP D2.

6 Conclusion and future work

In our previous work we have proposed an intelligent multiagent system based self-organising algorithm for multi-radio wireless mesh networks (MR-WMN) that can operate on any radio technology. The algorithm ensures scalability by progressively assigning the channels to nodes in clusters during the WMN system start up phase. The stability is offered by means of the proactive and reactive logic of the algorithm. These attributes were validated through analysis and simulation.

Through the work described in this report we have examined motivation and developed an algorithm for the topological control of MR-WMN. The goal of this algorithm is to increase the number of shortest paths to the portal nodes without adversely effecting interference cost. In addition to interference cost reduction implementation of this algorithm on MR-WMN further improve the system capacity.

Our future work will be focused on the development of our Java framework that is multi threaded so each node is represented as an independent thread. We believe that this will enable us to develop algorithms for tuning the capacity of the network links according to fluctuations in demand by mobile users.

References

1. Ko, B.J., Misra, V., Padhye, J., Rubenstein, D.: Distributed Channel Assignment in Multi-Radio 802.11 Mesh Networks. Technical report, Columbia University (2006)
2. Mishra, A., Rozner, E., Banerjee, S., Arbaugh, W.: Exploiting partially overlapping channels in wireless networks: Turning a peril into an advantage. In: ACM/USENIX Internet Measurement Conference. (2005)
3. Mishra, A., Shrivastava, V., Banerjee, S.: Partially Overlapped Channels Not Considered Harmful. In: SIGMetrics/Performance. (2006)
4. Akyildiz, I.F., Wang, X., Wang, W.: Wireless mesh netowrks: a survey. Computer Networks (2005) 445–487
5. Raniwala, A., Chiueh, T.c.: Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network. In: Proceedings IEEE Infocom '05, IEEE Computer Society (2005)
6. Vasudevan, S.: A Simulator for analyzing the throughput of IEEE 802.11b Wireless LAN Systems. Master's thesis, Virginia Polytechnic Institute and State University (2005)
7. Leith, D., Clifford, P.: A self-managed distributed channel selection algorithm for wlans. In: Proceedings of RAWNET, Boston, MA, USA (2006) 1–9
8. Duffy, K., Malone, D., Leith, D.: Modeling the 802.11 Distributed Coordination Function in Non-saturated Conditions. IEEE Communication Letters **9** (2005) 715–717
9. Tourrilhes, J.: Robust Broadcast: Improving the reliability of broadcast transmissions on CSMA/CA. In: Proceedings of PIMRC 1998. (1998) 1111–1115

Decisions with multiple simultaneous goals and uncertain causal effects

Paulo Trigo and Helder Coelho

Abstract A key aspect of decision-making in a disaster response scenario is the capability to evaluate multiple and simultaneously perceived goals. Current competing approaches to build decision-making agents are either mental-state based as BDI, or founded on decision-theoretic models as MDP. The BDI chooses heuristically among several goals and the MDP searches for a policy to achieve a specific goal. In this paper we develop a preferences model to decide among multiple simultaneous goals. We propose a pattern, which follows a decision-theoretic approach, to evaluate the expected causal effects of the observable and non-observable aspects that inform each decision. We focus on yes-or-no (i.e., pursue or ignore a goal) decisions and illustrate the proposal using the RoboCupRescue simulation environment.

1 Introduction

The mitigation of a large-scale disaster, caused either by a natural or a technological phenomenon (e.g., an earthquake or a terrorist incident), gives rise to multiple simultaneous goals that demand the immediate response of a finite set of specialized agents. In order to act rationally the agent must evaluate multiple and simultaneous perceived damages, account for the chance of mitigating each damage and establish a preferences relation among goals. The belief-desire-intention (BDI) mental-state architecture [7] is widely used to build reasoning agents, equipped with a set of beliefs about the state of the world and a set of desires which, broadly speaking, identify those states that the agent has as goals. From its beliefs and desires, and via

Paulo Trigo
Instituto Superior de Engenharia de Lisboa – ISEL, DEETC and LabMAg, GuIAA; Portugal
e-mail: ptrigo@deetc.isel.ipl.pt

Helder Coelho
Faculdade de Ciências da Universidade de Lisboa – FCUL, DI and LabMAg; Portugal
e-mail: hcoelho@di.fc.ul.pt

Please use the following format when citing this chapter:

Trigo, P. and Coelho, H., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 1322.

deliberation, the agent formulates an intention that can be seen as the goal, or desire, the agent commits to bring about. Although one side of rational behavior is the capability to establish preferences among simultaneous goals, current BDI theory and systems do not provide a theoretical or architectural framework for deciding how goals interact and how an agent decides which goals to pursue. When faced with multiple simultaneous goals, the intention selection (decision) follows a heuristic approach, usually coded by a human designer [4]. Additionally, BDI models find it difficult to deal with uncertainty, hence hybrid models have been proposed combining BDI and Markov decision process (MDP) approaches [5, 6]; however, hybrid models usually assume that the goal has already been chosen and tackle the stochastic planning problem (in order to achieve the chosen goal).

In this paper we take the decision-theoretic notion of rationality to estimate the importance of goals and to establish a preferences relation among multiple goals. We propose a preferences model that allows agent developers to design the relationships between perceived (certain) and uncertain aspects of the world in an easy and intuitive manner. The design is founded on the influence diagram [2] (ID) framework that combines uncertain beliefs and the expected gain of decisions. The proposal's practical usefulness is experimentally explored in a fire fighting scenario in the RoboCupRescue [3] domain. The decision model incorporates general fire fighting principles in a way that considerably simplifies the specification of a preferences relation among goals. Despite such simplification, the attained results are consistent with the initial fire fighting principles.

The next section describes the preferences model, which is instantiated and evaluated in section 3; the section 4 presents our conclusions and future goals.

2 The preferences model

The premisse of the preferences model is that the relation among simultaneous goals follows from the expected utility of the available decisions. The expected utility of a decision combines two elements: i) the value of the state under observation, and ii) the likelihood of success of that decision. Given a set of available decisions, $\mathscr{D}$, a set of states, $\mathscr{S}$, an utility function, $u : \mathscr{S} \rightarrow \mathbb{R}$, and the probability, $P(s \mid d)$, to achieve $s \in \mathscr{S}$ after decision $d \in \mathscr{D}$, the expected utility, $eu : \mathscr{D} \rightarrow \mathbb{R}$, of decision-making is described by: $eu(D = d) = \sum_{s \in \mathscr{S}} P(s \mid D = d)\, u(s)$, where D is a variable that holds an available decision. Given any goal there are always two available decisions: i) pursue the goal, or ii) ignore the goal. Thus, $\mathscr{D} = \{\texttt{yes}, \texttt{no}\}$, is such that $D_g = \texttt{yes}$ and $D_g = \texttt{no}$ represent, respectively, the decision to pursue or to ignore goal $g \in \mathscr{G}$.

The utility of a goal, g, measures the importance, assigned by the agent, to the goal g. The "importance" is a criterion related to a valuation in terms of benefits and costs an agent has of a mental state situation [1]. The mental state is materialized by the agent beliefs regarding the perceived states and the desire to pursue, or ignore, each goal. Also, the goal achievement payoff is estimated by the difference between the expected utility on pursuing and ignoring that goal. Thus, the goal utility func-

tion, $u_{\mathscr{G}}$, for each $g \in \mathscr{G}$, is defined by,

$$u_{\mathscr{G}}(g) = eu(D_g = \texttt{yes}) - eu(D_g = \texttt{no}) \tag{1}$$

The utility function, $u_{\mathscr{G}}$, is used to establish the preferences about the set of goals $\mathscr{G}$. The preferences, $\forall\, g_1, g_2 \in \mathscr{G}$, are: i) $g_1 \succ g_2$, if the agent prefers g_1 to g_2, or ii) $g_1 \sim g_2$, if the agent is indifferent between g_1 and g_2. The rules, used to establish the preferences' total order among goals, are described by,

$$g_1 \succ g_2 \begin{cases} u_{\mathscr{G}}(g_1) > u_{\mathscr{G}}(g_2) \\ \vee \\ u_{\mathscr{G}}(g_1) = u_{\mathscr{G}}(g_2) \ \wedge \ eu(D_{g_1} = \texttt{yes}) > eu(D_{g_2} = \texttt{yes}) \end{cases} \tag{2}$$

$$g_1 \sim g_2 \ \left\{ \ u_{\mathscr{G}}(g_1) = u_{\mathscr{G}}(g_2) \ \wedge \ eu(D_{g_1} = \texttt{yes}) = eu(D_{g_2} = \texttt{yes}) \right. \tag{3}$$

From expression 2 the agent prefers goals with higher payoff and when even, prefers goals that, when achieved, give higher expected advantage (i.e., higher $eu(D_g = \texttt{yes})$ value); the expression 3 indicates that in sight of equality the agent is indifferent between goals, thus taking, for instance, a random decision.

2.1 The causal effect pattern

The causal effects (consequences) of each decision are unknown, therefore our aim is to choose the decision alternative (goal) that minimizes the eventual disadvantageous consequences of such decision. The ID framework combines uncertain beliefs to compute the expected utility of decisions, thus rationality is a matter of choosing the alternative that leads to the highest expected utility, given the evidence of available information. The ID extends the, Bayesian network, chance nodes with two additional nodes: decisions and utilities, and two additional arcs: influences and informational. As in belief networks, chance nodes represent random variables, i.e., the agent's uncertain beliefs about the world. A decision node holds the available choices, i.e., the possible actions to take. An utility node represents the agent's preferences. The links between the nodes summarize their dependency relations.

We propose the following guidelines, to structure the multiple and simultaneous goals decision problem, using the ID framework:

i. the current state is characterized by a set of variables that are observable at the decision-making time instant,
ii. the decision outcome is characterized by a set of variables that are non-observable at the decision-making time instant,
iii. the observable variables inform the decision nodes and the decision nodes influence the non-observable variables,
iv. the observable variables condition the non-observable variables,

v. all dependencies among observable variables, or among non-observable variables are valid (whilst not generating any dependency loop),
vi. the set of observable variables influences a set of utility nodes,
vii. the set of non-observable variables influences a set of utility nodes,
viii. the two sets of utility nodes (cf. items vi vii) are disjoint, and
ix. a decision influences both sets of utility nodes (cf. items vi vii).

Figure 1 illustrates the above guidelines using the regular ID symbols; circle is a chance node, rectangle is a decision node and the lozenge is an utility node.

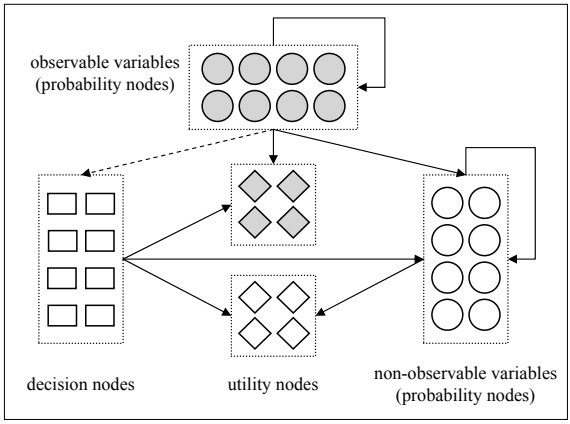


Fig. 1 The influence diagram (ID) pattern (sets are represented by dotted rectangles; gray elements refer to observable information; dotted arcs are informational and the other are conditional arcs).

The gray filling (cf. figure 1) has special meaning: i) the gray chance node indicates information availability, i.e., an observable variable (cf. item i above), and ii) the gray utility node indicates a dependency from a gray chance node, i.e., the utility of some observable variables (cf. item vi above). The sets of nodes with similar characteristics are aggregated by a dotted rectangle. The arcs connect sets of nodes (instead of individual nodes), therefore attaining an ID pattern, i.e., a template from which to build several different instances with the same overall structure.

2.2 The ID pattern usage

The ID pattern (cf. figure 1) is used to support the construction of the goal utility function, $u_{\mathscr{G}}$ (cf. equation 1). Therefore, we propose the following method to specify the decision nodes:

i. identify the largest subsets of goals, $\mathscr{G}_i \subseteq \mathscr{G}$ such that $\cup_i \mathscr{G}_i = \mathscr{G}$ and all the goals $g \in \mathscr{G}_i$ are characterized by the same set of observable variables,

ii. for each $\mathscr{G}_i$ (cf. item i) specify a decision node labeled "D_i" and add the corresponding information arcs (from observable variables to "D_i"),
iii. for each decision node, "D_i", set its domain to "`yes`" and "`no`" values to represent, respectively, the decision to pursue, or ignore, a goal $g \in \mathscr{G}_i$; the goal, g, occurs after the observation of the variables that inform "D_i".

For concreteness and to illustrate the design of the decision problem, the next section materializes the preferences model in a simulated scenario.

3 Experimental setup

We used the RoboCupRescue environment to devised a disaster scenario that evolves at the Nagata ward in Kobe, Japan. Two buildings, B_1 and B_2, not far from each other (about 90 meters) catch a fire. The B_1 is relatively small and is located near Kobe's harbor, in a low density neighborhood. The B_2 is of medium size and it is highly surrounded by other buildings. As time passes, the fires' intensity increase so a close neighbor is also liable to catch a fire.

Figure 2 shows the disaster scenario; each opaque rectangle is a building and a small circle is positioned over B_1 and B_2. The two larger filmy squares define the neighborhood border of B_1 and B_2 within a d distance (in meters). The ground neighborhood area of a building is given by $ngb(d) = (2 \times d)^2$, for a distance d, and the set of buildings contained within such area is denoted as $\mathscr{N}_{B_i,d}$; we set $d = 250$ m, thus a ground neighborhood area of 250.000 m^2.

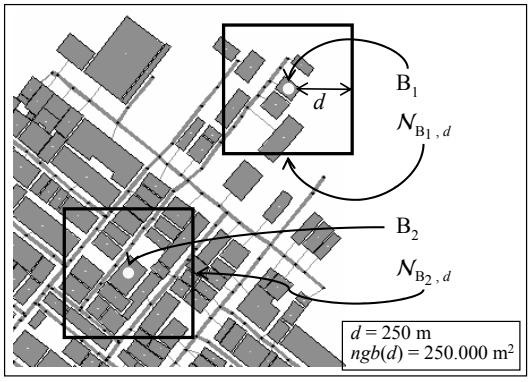


Fig. 2 Fire scenario in the buildings labeled B_1 and B_2 (the set of buildings contained within each building's neighborhood, $ngb(d)$, is represented by $\mathscr{N}_{B_i,d}$).

To simplify we assume that: i) buildings use identical construction materials, ii) buildings are residential (neither offices nor industries inside the buildings), and iii) there are no civilians, caught by fires, inside the buildings. We also assume that agents get informed about the fires the moment it starts; we are not concerned on how (through which communication channel) the agent gets such information. We

now carry on with the design of the multiple simultaneous goal decision problem, in the context of this illustrative domain.

3.1 The ID pattern instance

In order to apply the ID pattern (cf. figure 1) to the illustrative scenario (cf. figure 2) we considered, for each building, the following observable variables:

- The building's fire intensity, *fireIntensity* (we adopted the RoboCupRescue names), perceived by the agent with three values: i) 1, an early fire, ii) 2, an increasing intensity fire, and iii) 3, a high intensity fire.
- The building's total area, *allFloorsArea*, given by the building's ground area times the number of floors, with three values: i) `low`, ii) `medium`, and iii) `high`. Each value is $\frac{1}{3}$ of the normalized total area, i.e., the building's total area divided by the maximum total area of the buildings in the scenario ; e.g. for B_1 we have $\frac{7.668}{57.866} = 0{,}13$ (`low`) and for B_2 we have $\frac{19.454}{57.866} = 0{,}34$ (`medium`).
- The building's neighborhood density, *neighbourhoodDensity*, computed as the ratio between the summation of the ground area, *floorArea*(b), of each building within distance d of B_i neighborhood (i.e., each $b \in \mathcal{N}_{B_i,d}$), and the total area of that neighborhood (i.e., $ngb(d)$); the ratio is thus given by, $\frac{\sum_{b \in \mathcal{N}_{B_i,d}} floorArea(b)}{ngb(d)}$), and the *neighbourhoodDensity* has the following three values: i) `low`, ii) `medium`, and iii) `high`. Each value is $\frac{1}{3}$ of that ratio; e.g. for B_1 we have $\frac{39.900}{250.000} = 0{,}16$ (`low`) and for B_2 we have $\frac{138.500}{250.000} = 0{,}55$ (`medium`).

The non-observable variable, *destruction*, describes the destruction inflicted by the fire with three values, `low`, `medium`, and `high`, each representing, respectively, the intervals $[0;0{,}2]$, $[0{,}2;0{,}7]$ and $[0{,}7;1]$ of the destruction percentage.

The goals are $extinguished(B) \in \mathcal{G}' \subseteq \mathcal{G}$, where B is a building in fire. For readability, the subset $\mathcal{G}'$ will be named as *extinguish*. Hence, we specify a decision variable, *extinguish* (cf. section 2.2), that evaluates each goal, $extinguished(B)$, whereas all the aspects that influence the decision (extinguish or ignore the fire in B), are represented through the observable variables: *fireIntensity*, *allFloorsArea* and *neighbourhoodDensity*.

To specify the utility nodes we follow *three general fire attack strategies* that, although intuitive, were acquired after the RoboCupRescue experimentation:

- the earlier a fire is attacked, the easier it is to extinguish the fire,
- the smaller the building, the less time it takes to extinguish the fire, and
- the higher the neighborhood density, the higher the need to extinguish the fire.

The above strategies are used to specify the utility nodes: *U1* and *U2*. The utility node *U1* is influenced by the observable variables and represents the agent's evaluation of the fire intensity impact on the neighbor building. For example, a fire may

cause higher damages in a high density than in a low density neighborhood (given an identical fire intensity); thus, the higher utility values are ascribed to high intensity fires that occur in high density neighborhoods. The utility node *U2* is influenced by the non-observable variable and represents the agent's evaluation of the building's expected final destruction. For example, an early fire is expected to cause a lower destruction than a high intensity fire (given equivalent total areas and neighborhood density); thus, higher utility values are ascribed to early low intensity fires.

Figure 3 presents the ID that assembles all the above analysis: observable and non-observable variables, decision and utility nodes.

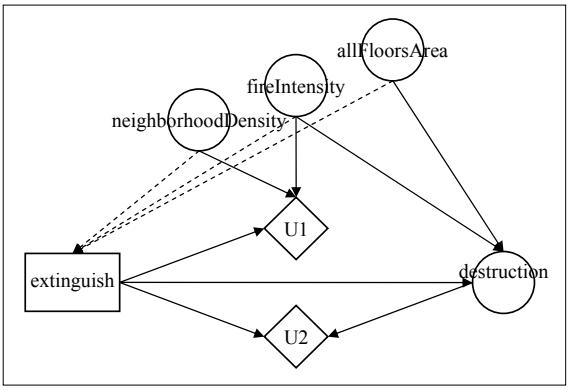


Fig. 3 Influence diagram for the *extinguish* set of goals (the construction follows the ID pattern, depicted in figure 1, thus adopting the terminology thereby defined).

The ID (cf. figure 3) is an instance of the proposed causal effect pattern (cf. figure 1) and digests the analysis of the illustrative scenario (cf. figure 2). The figure 3 intelligibility stresses that the ID is very handy in revealing the structure (the influence among the decision constituents) of the decision problem.

3.2 The preferences relation

After the ID structure we built the conditional probability and utility tables (CPT and UT) attached, respectively, to each chance and utility node. The CPT represents the probabilistic knowledge about the causal relations among the state variables. The UT specifies a decision-making strategy.

Our strategy follows the *three general fire attack strategies*. Figure 4 shows the *extinguish* expected utility, and each situation is represented by a vector,

$$v \equiv \langle neighbourhoodDensity, allFloorsArea, fireIntensity \rangle ,$$

of perceived values (of the observable variables), for each building with a fire.

Each v variable is graphically discriminated as: i) the *neighbourhoodDensity* is a circle that becomes larger as the neighborhood density increases, ii) circles in

vertical lines follow the *allFloorsArea* value, upper circles having lower areas, and iii) the clustering of *fireIntensity* is marked in the graphic. For example, the B_1 and B_2 vectors are, respectively, $\langle \texttt{low}, \texttt{low}, 1 \rangle$ and $\langle \texttt{medium}, \texttt{medium}, 1 \rangle$.

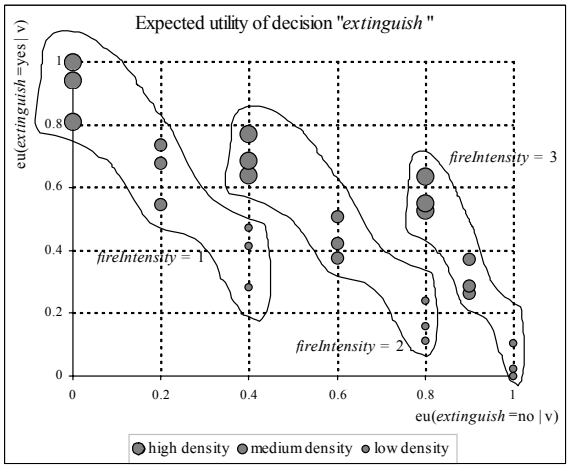


Fig. 4 Decision *extinguish*, given the observation of *v* (*neighbourhoodDensity*, *allFloorsArea* e *fireIntensity*); the buildings B_1 and B_2 are labeled in the graphic.

The expected *extinguish* utility, given *v* observation, is depicted in the abscissa and ordinate axis (cf. figure 4), respectively, by $eu(\, extinguish_g = \texttt{no} \mid v\,)$ and $eu(\, extinguish_g = \texttt{yes} \mid v\,)$, where subscript, *g*, is the goal of extinguishing the fire in the building where *v* was perceived, i.e., *g* represents the desired situation for the building (for readability symbol *g* is not plotted in figure 4).

Figure 4 shows that as the fire intensity increases the utility to extinguish decreases and the utility to ignore the fire increases although being interleaved with the various neighborhood densities (thus accounting the fire spreading effect). The highest utility (to extinguish) is assigned to lower area buildings (given equal values for other variables) as they are simpler and faster to control.

Figure 5 plots the agent preferences (given by expressions 2 and 3); a small diamond represents each previously shown *v* instance (cf. figure 4); a line goes through all diamonds in a course that links two adjacent priority situations (the darker segment highlights the B_2 to B_1 path). The highest preference is $\langle \texttt{high}, \texttt{low}, 1 \rangle$, i.e., an early fire in a small building in a high density neighborhood; the lowest preference is $\langle \texttt{low}, \texttt{high}, 3 \rangle$, i.e., a high intensity fire in a big building in a low density neighborhood. Table 1 details B_2 to B_1 preferences and shows that the three early fires are interleaved with higher intensity fires located in increasing density neighborhoods or decreasing area buildings. It is also interesting to note that the 2^{nd} and 5^{th} buildings only differ in their dimension (*allFloorsArea*) and the two buildings between them (3^{rd} and 4^{th}) have increasing neighborhood density and fire intensity and decreasing total area.

The interleaving of situations (shown in figure 5 and detailed in table 1), represents the trade-off, obtained from applying the expression 2, among our *three ge-*

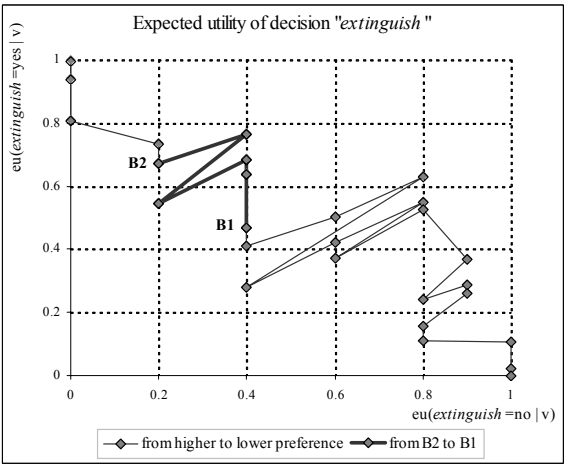


Fig. 5 A preference relation for the decision *extinguish*.

Table 1 Preferences detail (from B_2 to B_1); the first column identifies the buildings and the last column shows the utility value $u_{\mathscr{G}} \in [0,1]$; the first line represents B_2.

	Order	*neighbourhoodDensity*	*allFloorsArea*	*fireIntensity*	$\frac{u_{\mathscr{G}} - \min u_{\mathscr{G}}}{\max u_{\mathscr{G}} - \min u_{\mathscr{G}}}$
B_2	1^{st}	medium	medium	1	0,73
...	2^{nd}	high	low	2	0,69
	3^{rd}	medium	high	1	0,66
	4^{th}	high	medium	2	0,65
...	5^{th}	high	high	2	0,62
B_1	6^{th}	low	low	1	0,53

neral fire attack strategies. The rationality of those strategic guidelines may be disputed by a domain specialist for their verisimilitude with the real-world fire brigade strategies. Such dispute is a relevant contribution to adjust and mature the ID design but it is not the central discussion of this paper.

3.3 The decision design complexity

To apply the *three general fire attack strategies* (or any strategy set) a human designer would traduce its rationality into a total order relation among the state space. However, building a total order quickly becomes too complex. For example, our illustrative scenario has 4 variables, each with 3 values, thus a total of $3^4 = 81$ situations. To establish a total order, the human must compare each situation with all the others; in the worst case $\sum_{i=1}^{81-1} i = \frac{80 \times 81}{2} = 3240$ comparisons; in the best case (if able to apply a divide-and-conquer method), $81 \times \log_2 81 = 514$ comparisons. It is not likely that a human designer fulfils all those comparisons to establish a total order among all possible situations.

Our proposed ID design is much simpler. Assign, to each decision, the utility of observable and non-observable variables: $3^2 \times 2 + 3 \times 2 = 24$ assignments. This is an important complexity reduction: about 95% (from 514 to 24) in the above best case and about 99% (from 3240 to 24) in the above worst case. Despite that reduction the results (cf. figure 5 and table 1) exhibit a plausible translation of the general strategies used to guide the decision model design.

4 Conclusions and future work

This paper addresses a shortcoming, of current work, in the design of agents that act in complex domains: the evaluation of multiple simultaneous goals with observable and non-observable world state aspects. We propose a pattern, based on the influence diagram framework, to specify both the uncertainty of causal effects and the expected gain with regard to the decision of whether to pursue or ignore each goal. Practical experiences indicate that the ID pattern considerably simplifies the specification of a decision model (in RoboCupRescue domain) and enabled to established a preferences order among goals that is consistent with the initial, domain expert, very general strategies. This work represents the ongoing steps in a line of research that aims to develop decision-making agents that inhabit complex environments (e.g., the RoboCupRescue). Future work will apply the preferences model to the problem of coordinating teamwork (re)formation [6] from a centralized perspective.

Acknowledgements This research was partially supported by the LabMAG FCT research unit.

References

1. Corrêa, M., Coelho, H.: Collective mental states in extended mental states framework. In: Proceedings of the IV International Conference on Collective Intentionality. Certosa di Pontignano, Siena, Italy (2004)
2. Howard, R., Matheson, J.: Influence diagrams. In: Readings on the Principles and Applications of Decision Analysis, vol. 2, pp. 721–762. Strategic Decision Group, Menlo Park, CA (1984)
3. Kitano, H., Tadokoro, S.: RoboCup Rescue: A grand challenge for multi-agent systems. Artificial Intelligence Magazine **22**(1), 39–52 (2001)
4. Pokahr, A., Braubach, L., Lamersdorf, W.: A goal deliberation strategy for BDI agent systems. In: Proceedings of the Third German Conference on Multi-Agent System Technologies (MATES-2005), pp. 82–94. Springer (2005)
5. Simari, G., Parsons, S.: On the relationship between MDPs and the BDI architecture. In: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-06), pp. 1041–1048. ACM Press, Hakodate, Japan (2006)
6. Trigo, P., Coelho, H.: Decision making with hybrid models: the case of collective and individual motivations. In: Proceedings of the EPIA-07 International Conference (New Trends in Artificial Intelligence), pp. 669–680. Guimarães, Portugal (2007)
7. Wooldridge, M.: Reasoning About Rational Agents, chap. Implementing Rational Agents. The MIT Press (2000)

Agent Based Frequent Set Meta Mining: Introducing EMADS

Kamal Ali Albashiri, Frans Coenen, and Paul Leng

Abstract In this paper we: introduce EMADS, the Extendible Multi-Agent Data mining System, to support the dynamic creation of communities of data mining agents; explore the capabilities of such agents and demonstrate (by experiment) their application to data mining on distributed data. Although, EMADS is not restricted to one data mining task, the study described here, for the sake of brevity, concentrates on agent based Association Rule Mining (ARM), in particular what we refer to as frequent set meta mining (or Meta ARM). A full description of our proposed Meta ARM model is presented where we describe the concept of Meta ARM and go on to describe and analyse a number of potential solutions in the context of EMADS. Experimental results are considered in terms of: the number of data sources, the number of records in the data sets and the number of attributes represented.

Keywords: Multi-Agent Data Mining (MADM), Frequent Itemsets, Meta ARM, Association Rule Mining.

1 Introduction

In this paper an extendible multi-agent data mining framework that can enable and accelerate the deployment of practical solutions to data mining problems is introduced. The vision is a collection of data, data mining and user agents operating under decentralised control. Practitioners wishing to participate in the framework may add additional agents using a registration strategy. We envision a collection of scattered data over the network, accessed by a group of agents that allow a user to pose data mining queries to those data sources with no requirement to know the location of the supporting data, nor how the agents are materialised through an inte-

Kamal Ali Albashiri, Frans Coenen, and Paul Leng
Department of Computer Science, The University of Liverpool,
Ashton Building, Ashton Street, Liverpool L69 3BX, United Kingdom
e-mail: {ali,frans,phl}@csc.liv.ac.uk

Please use the following format when citing this chapter:

Albashiri, K.A., Coenen, F. and Leng, P., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 2332.

gration and ranking process. We also envision that the inclusion of a new data source or data mining techniques should be a simple process of adding new agents to the system. To investigate the potential of this approach we have built EMADS (Extendible Multi-Agent Data mining System). The use of EMADS offers a number of advantages, includes: decentralised control, distribution of computational resources, interoperability, distribution of expertise, task and data matching, Result evaluation, simple extendibility and security.

To illustrate some of the features of EMADS a Meta ARM (Association Rule Mining) scenario is considered in this paper. We define the term Meta Mining as the process of combining the individually obtained results of N applications of a data mining activity. The motivation behind the scenario is that data relevant to a particular ARM application is often owned and maintained by different, geographically dispersed, organizations. Information gathering and knowledge discovery from such distributed data sources typically entails a significant computational overheads; computational efficiency and scalability are both well established critical issue in data mining [1]. One approach to addressing problems such as the meta ARM problem is to adopt a distributed approach. However this requires expensive computation and communication costs. In distributed data mining, there is a fundamental tradeoff between accuracy and cost of computation. If we wish to improve the computation and communication costs, we can process all the data locally obtaining local results, and combine these results centrally to obtain the final result. If our interest is in the accuracy of the result, we can ship all the data to a single node (and apply an appropriate algorithm to produce this desired result). In general the latter is more expensive while the former is less accurate. The distributed approach also entails a critical security problem in that it reveals private information; privacy preserving issues [2] are of major concerns in inter enterprise data mining when dealing with private databases located at different sites.

An alternative approach to distributed data mining is high level learning which adopts strategies to allow all data to be locally analyzed, local results (models) are then combined at a central site to obtain the final result (global model). This approach is less expensive but may produce ambiguous and in- correct global results. To make up for such a weakness, many researchers have attempted to identify further alternatives to combining local models built at different sites. Most of these approaches are agent-based high level learning strategies such as: meta-learning [4], mixture of experts [5] and knowledge probing [6]. Bagging [7] increases the accuracy of the model by generating multiple models from different data sets chosen uniformly with replacement and then averaging the outputs of the models. However, these approaches still only have the ability to estimate a global data model through the aggregation of the local results, rather than generating an exact correct global model.

In EMADS a distributed computation framework is defined in terms of a Multi-Agent System (MAS), i.e. a system composed of a community of agents, capable of reaching goals that are difficult to achieve by an individual system [3]. In addition, a MAS can display self-organizational and complex behaviours, even when the capabilities of individual agents are relatively simple. The fundamental distinc-

tion between a distributed architecture and a MAS architecture is one of control. In a distributed system control is centralized; in a MAS control is decentralized in that agents are self motivating, and problem solving is achieved through inter-communication between agents.

The rest of this paper is organised as follows. Section 2 provides the motivation behind the material presented and discusses some related work. For completeness a brief note on Meta ARM Algorithms is then presented in Section 3. In section 4 our Meta ARM model architecture and functionality are described. Section 5 discusses the experimental results. Finally, Section 6 concludes the paper.

2 Related Work

There are a number of reports in the literature of the application of Agent techniques to data mining. Kargupta, Stafford, and Hamzaoglu [11] describe a parallel data mining system (PADMA) that uses software agents for local data accessing and analysis, and a Web based interface for interactive data visualization. PADMA has been used in medical applications. The meta-learning strategy offers a way to mine classifiers from homogeneously distributed data. Perhaps the most mature systems of agent-based meta-learning systems are: JAM [4], BODHI [12], and Papyrus [13]. In contrast to JAM and BODHI, Papyrus can not only move models from site to site, but can also move data when that strategy is desired. Papyrus is a specialized system which is designed for clustering while JAM and BODHI are designed for data classification. Basically, these systems try to combine local knowledge to optimize a global objective.

The major criticism of such systems is that it is not always possible to obtain an exact final result, i.e. the global knowledge model obtained may be different from the one that might have been obtained by applying the one model approach to the same data.

3 Note on Meta ARM Algorithms

Association Rule Mining (ARM) is concerned with the identification of patterns (expressed as "if ... then ..." rules) in data sets [8]. ARM typically begins with the identification of *frequent sets* of data attributes that satisfy threshold requirements of *relative* support in the data being examined. The most significant issue when combining groups of previously identified frequent sets is that wherever an itemset is frequent in a data source A but not in a data source B a check for any contribution from data source B is required (so as to obtain a global support count). The challenge is thus to combine the results from N different data sources in the most computationally efficient manner. This in turn is influenced predominantly by the magnitude (in terms of data size) of returns to the source data that are required.

To investigate and evaluate our ideas on EMADS a study of Meta ARM is presented here. Five Meta ARM algorithms are considered, all founded on the well known TFP ARM algorithm [9, 10] where results are stored in a T-tree. For the Meta ARM these trees must then be merged in some way. The structure of the T-tree, and the algorithms used in its construction, are described in [10]; the details of this are not relevant to the present paper, and in principle any algorithm for generating frequent sets could have been employed. As with all such algorithms, the merging of locally frequent sets to produce global totals may require additional computation to complete the counts of some sets. Each of the Meta ARM algorithms/agents makes use of *return to data* (RTD) lists, at least one per data set/agent, to hold lists of itemsets whose support was not included in the current T-tree and for which the count is to be obtained by a return to the originating raw data agent. The processing of RTD lists may occur during, and/or at the end of, the Meta ARM process depending on the nature of the algorithm. The algorithms can be summarised as follows:

1. Brute Force: Merges the T-trees one by one starting with the largest tree generating (N) RTD lists, processes RTD lists and prunes the T-tree at end of the merge process.
2. Apriori: Merges all T-trees level by level starting from the first level ($K = 1$) generating ($K * N$) RTD lists, processes RTD lists and prunes the T-tree at each level. The objective of the Apriori Meta ARM algorithm is to identify unsupported itemsets earlier in the process.
3. Hybrid 1: Commences by generating the top level of the merged T-tree in the Apriori manner described above (including processing of the RTD list); and then adds the appropriate branches, according to which top level nodes are supported, using a Brute Force approach.
4. Hybrid 2: Commences by generating the top two levels of the merged T-tree, instead of only the first level, as in the Hybrid 1 approach. Additional support counts are obtained by processing the RTD lists. The remaining branches are added to the supported level 2-nodes in the merged T-tree sofar (again) using the Brute Force mechanism.
5. Bench Mark: It is a bench mark algorithm against which the identified Meta ARM algorithms were to be compared.

Full details of the Meta ARM algorithms can be found in Albashiri et al. ([14]). Note that the overview given here is in the context of MADM (Multi-Agent Data Mining) whereas the original algorithms proposed by Albashiri et al. did not operate in an agent context.

4 Meta ARM Model

In order to demonstrate the feasibility of our EMADS vision a peer to peer agent-based framework has been designed and implemented, which uses a broker mediated-based architectural model [15].

Fig. 1 shows the Meta ARM model architecture of EMADS framework which is built with the JADE Toolkit [16]. The system consists of one organization site (mediator host) and several local sites (sites of individual hosts). Detailed data are stored in the DBMS (Data Base Management System) of local sites. Each local site has at least one agent that is a member of the organization. The connection between a local agent and its local DBMS is not included. There are two special JADE agents at the organization site (automatically started when the organization site is launched). The AMS (Agent Management System) provides the Naming Service (i.e. ensures that each agent in the platform has a unique name) and represents the authority in the platform. The DF (Directory Facilitator) provides a Yellow Pages service by means of which an agent can find other agents providing the services required in order to achieve its goals. All Routine communication and registration are managed by these JADE agents. Data mining tasks are managed through the P2P model by the other agents.

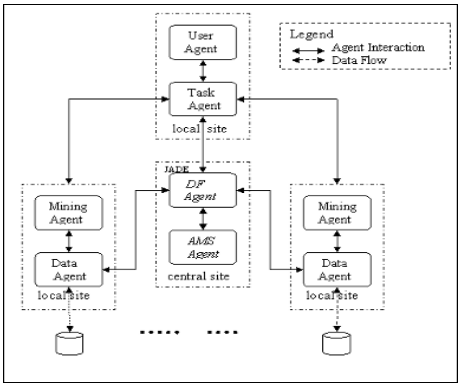


Fig. 1. Meta ARM Model Architecture

In this framework, agents are responsible for accessing local data sources and for collaborative data analysis. The architecture includes: (i) data mining agents, (ii) data agents, (iii) task agents, (iv) user agents, and (v) mediators (JADE agents) for agents coordination. The data and mining agents are responsible for data accessing and carrying through the data mining process; these agents work in parallel and share information through the task agent. The task agent co-ordinates the data mining operations, and presents results to the user agent. Data mining is carried out by means of local data mining agents (for reasons of privacy preservation). In the context of Meta ARM activity each local mining agent's basic function is to generate local item sets (local model) from local data and provide this to the task agent in order to generate the complete global set of frequent itemsets (global model).

4.1 Dynamic Behaviour of System for Meta ARM operations

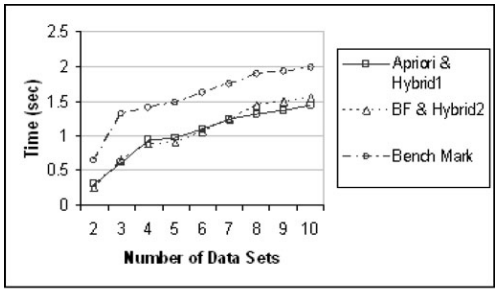
The system Initially starts up with the two central JADE agents. When a data agent wishes to make its data available for possible data mining tasks, it must publish its name and description with the DF agent. In the context of Meta ARM, each mining agent could apply a different data mining algorithm to produce its local frequent item sets T-tree. The T-trees from each local data mining agent are collected by the task agent, and used as input to Meta ARM algorithms for generating global frequent item sets (merged T-tree) making use of return to data (RTD) lists, at least one per data set, to contain lists of itemsets whose support was not included in the current T-tree and for which the count is to be obtained by a return to the raw data.

5 Experimentation and Analysis

To evaluate the five Meta ARM algorithms, in the context of EMADS vision, a number of experiments were conducted. These are described and analysed in this section. The experiments were designed to analyse the effect of the following:

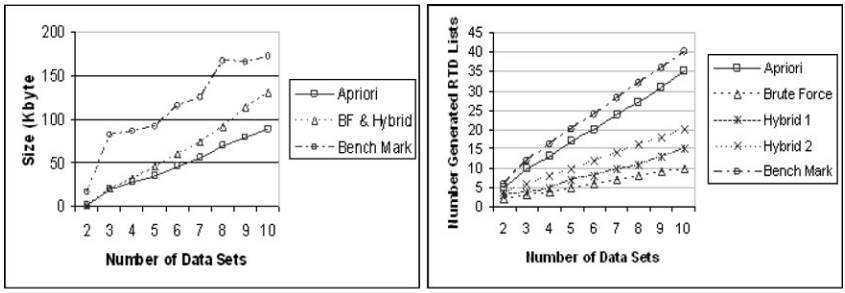
1. The number of data sources *(data agents)* .
2. The size of the datasets (held at data agents) in terms of number of records.
3. The size of the datasets (held at data agents) in terms of number of attributes.

Experiments were run using two Intel Core 2 Duo E6400 CPU (2.13GHz) computers with 3GB of main memory (DDR2 800MHz), Fedora Core 6, Kernel version 2.6.18 running under Linux except for the first experiment where two further computers running under Windows XP were added. For each of the experiments we measured: (i) processing time (seconds/mseconds), (ii) the size of the RTD lists (Kbytes) and (iii) the number of RTD lists generated. The authors did not use the IBM QUEST generator [17] because many different data sets (with the same input parameters) were required and it was found that the quest generator always generated the same data given the same input parameters. Instead the authors used the LUCS KDD data generator [1]. Note that the slight oscillations in the graphs result simply from a vagary of the random nature of the test data generation.



(a) Processing Time

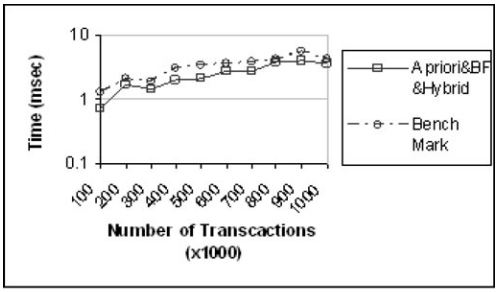
[1] *http://www.csc.liv.ac.uk/frans/KDD/Software//LUCS-KDD-DataGen/*



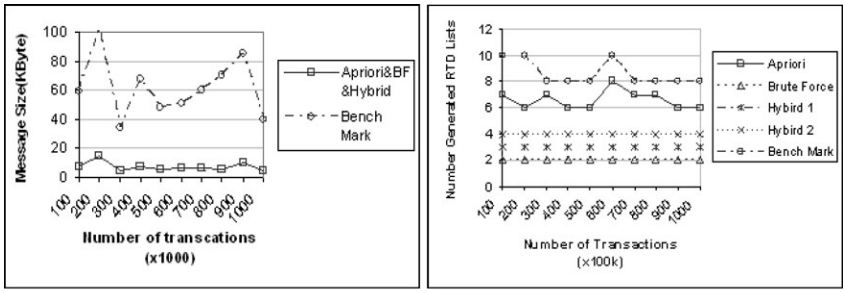
(b) Total size of RTD lists (c) Number of RTD lists

Fig. 2. Effect of number of data sources

Figure 2 shows the effect of adding additional data sources. For this experiment ten different artificial data sets were generated and distributed among four machines using $T = 4$ (average number of items per transactions), $N = 20$ (Number of attributes), $D = 100k$ (Number of transactions). The selection of a relatively low value for N ensured that there were some common frequent itemsets shared across the T-trees. Experiments using $N = 100$ and above tended to produce many frequent 1-itemsets, only a few isolated frequent 2-itemsets and no frequent sets with cardinality greater than 2. For the experiments a support threshold of 1% was selected. Graph 2(a) demonstrates that all of the proposed Meta ARM algorithms worked better then the bench mark (start from "scratch") approach. The graph also shows that the Apriori Meta ARM algorithm, which invokes the "return to data procedure" many more times than the other algorithms, at first takes longer; however as the number of data sources increases the approach starts to produce some advantages as T-tree branches that do include frequent sets are identified and eliminated early in the process. The amount of data passed to and from sources, shown in graph 2(b), correlates directly with the execution times in graph 2(a). Graph 2(c) shows the number of RTD lists generated in each case. The Brute Force algorithm produces one (very large) RTD list per data source. The Bench Mark algorithm produces the most RTD lists as it is constantly returning to the data sets, while the Apriori approach produces the second most (although the content is significantly less).



(a) Processing Time

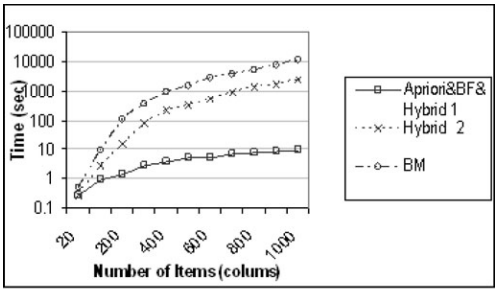


(b) Total size of RTD lists (c) Number of RTD lists

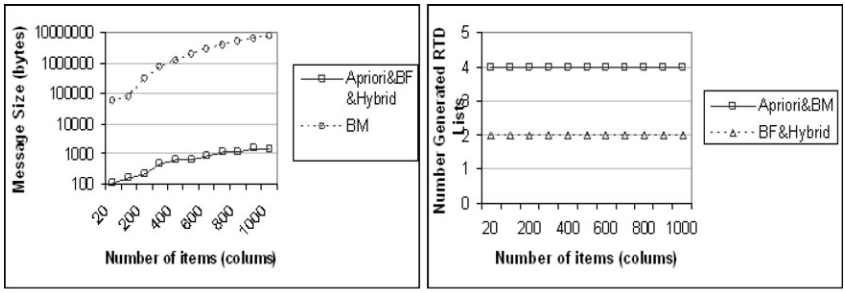
Fig. 3. Effect of increasing number of records

Figure 3 demonstrates the effect of increasing the number of records. The input data for this experiment was generated by producing a sequence of ten pairs of data sets (with $T = 4, N = 20$) representing two sources on two different machines. From graph 3(a) it can be seen that the Brute Force and Hybrid 1 algorithms work best because the size of the return to data lists are limited as no unnecessary candidate sets are generated. This is illustrated in graph 3(b). Graph 3(b) also shows that the increase in processing time in all cases is due to the increase in the number of records only; the size of the RTD lists remains constant throughout as does the number of RTD lists generated (graph 3(c)).

Figure 4 shows the effect of increasing the global pool of potential attributes (remember that each data set will include some subset of this global set of attributes). For this experiment another sequence of pairs of data sets (representing two sources) was generated with $T = 4$, $D = 100K$ and N ranging from 100 to 1000. As in the case of experiment 2 the Brute Force and Hybrid 1 algorithms work best (for similar reasons) as can be seen from graph 4(a). However in this case (compared to the previous experiment), the RTD list size did increase as the number of items increased (graph 4(b)). For completeness graph 4(c) indicates the number of RTD lists sent with respect to the different algorithms. The reasoning behind the Hybrid 2 algorithm proved to be unfounded; all the 1-itemsets tended not to be all supported.



(a) Processing Time



(b) Total size of RTD lists (c) Number of RTD lists

Fig. 4. Effect of increasing number of items (attributes)

All the Meta ARM algorithms outperformed the bench mark (start from scratch) algorithm. The Hybrid 2 algorithm performed in an unsatisfactory manner largely because of the size of the RTD lists sent. Of the remainder the Apriori approach coped best with a large number of data sources, while the Brute Force and Hybrid 1 approaches coped best with increases data sizes (in terms of column/rows) again largely because of the relatively smaller RTD list sizes. It should also be noted that the algorithms are all complete and correct, i.e. the end result produced by all the algorithms is identical to that obtained from mining the union of all the raw data sets using some established ARM algorithm. Of course our MADM scenario, which assumes that data cannot be combined in this centralised manner, would not permit this.

6 Conclusions and Future Work

Traditional centralized data mining techniques may not work well in many distributed environments where data centralization may be difficult because of limited bandwidth, privacy issues and/or the demand on response time. Meta-learning data mining strategies may offer a better solution than the central approaches but are not as accurate in their results. This paper proposes EMADS, multi-agent data mining framework with peer-to-peer architecture as an application domain to address the above issues. The use of EMADS was illustrated using a meta ARM scenario. Four meta ARM algorithms and a bench mark algorithm were considered. The described experiments indicated, at least with respect to Meta ARM, that EMADS offers positive advantages in that all the Meta ARM algorithms were more computationally efficient than the bench mark algorithm. The results of the analysis also indicated that the Apriori Meta ARM approach coped best with a large number of data sources, while the Brute Force and Hybrid 1 approaches coped best with increased data sizes (in terms of column/rows). The authors are greatly encouraged by the results obtained so far and are currently undertaking further analysis of EMADS with respect to alternative data mining tasks.

References

1. Kamber, M., Winstone, L., Wan, G., Shan, S. and Jiawei, H., "Generalization and Decision Tree Induction: Efficient Classification in Data Mining". Proc. of the Seventh International Workshop on Research Issues in Data Engineering, pp. 111-120, 1997.
2. Aggarwal, C. C. and Yu, P. S., "A Condensation Approach to Privacy Preserving Data Mining". Lecture Notes in Computer Science, Vol. 2992, pp. 183-199, 2004
3. Wooldridge, M., "An Introduction to Multi-Agent Systems". John Wiley and Sons Ltd, paperback, 366 pages, ISBN 0-471-49691-X, 2002.
4. Stolfo, S., Prodromidis, A. L., Tselepis, S. and Lee, W., "JAM: Java Agents for Meta-Learning over Distributed Databases". Proc. of the International Conference on Knowledge Discovery and Data Mining, pp. 74-81, 1997.
5. Xu, L. and Jordan, M. I., "EM learning on a generalised finite mixture model for combining multiple classifiers", In Proc. of World Congress on Neural Networks, 1993.
6. Guo, Y. and Sutiwaraphun, J., "Knowledge probing in distributed data mining". In Advances in Distributed and Parallel Knowledge Discovery, 1999.
7. Breiman, L., Bagging predictors, Machine Learning, 24, 123-140, 1996.
8. Agrawal, R., Imielinski, T., and Swami A., "Mining Association Rules between Sets of Items in Large Databases". In Proc. of ACM SIGMOD Conference on Management of Data, Washington DC, May 1993.
9. Goulbourne, G., Coenen, F.P. and Leng, P., "Algorithms for Computing Association Rules Using A Partial-Support Tree". Proc. ES99, Springer, London, pp. 132-147, 1999.
10. Coenen, F.P. Leng, P., and Goulbourne, G., "Tree Structures for Mining Association Rules". Journal of Data Mining and Knowledge Discovery, Vol 8, No 1, pp. 25-51, 2004.
11. Kargupta, H., Hamzaoglu, I., and Stafford, B., "Scalable, Distributed Data Mining Using an Agent Based Architecture". Proc. of Knowledge Discovery and Data Mining, AAAI Press, 211-214, 1997.
12. Kargupta, H., Hersh berger, D., and Johnson, E., Collective Data Mining: A New Perspective Toward Distributed Data Mining. Advances in Distributed and Parallel Knowledge Discovery, MIT/AAAI Press, 1999.
13. Bailey S., Grossman, R., Sivakumar, H., and Turinsky, A., "Papyrus: a system for data mining over local and wide area clusters and super-clusters". In Proc. Conference on Supercomputing, page 63. ACM Press, 1999.
14. Albashiri, K.A., Coenen, F.P., Sanderson, R. and Leng. P., Frequent Set Meta Mining: Towards Multi-Agent Data Mining. To appear in Research and Development in Intelligent Systems XXIV, Springer, London, (proc. AI'2007).
15. Schollmeier, R., "A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications". First International Conference on Peer-to-Peer Computing (P2P01) IEEE. August 2001.
16. Bellifemine, F., Poggi, A., and Rimassi, G., "JADE: A FIPA-Compliant agent framework". Proc. Practical Applications of Intelligent Agents and Multi-Agents, April 1999, pg 97-108 (See http://sharon.cselt.it/projects/jade for latest information)
17. Agrawal, R., Mehta, M., Shafer, J., Srikant, R., Arning, A. and Bollinger, T.,"The Quest Data Mining System". Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining, (KDD1996).

AGENTS 2

On the evaluation of MAS development tools

Emilia Garcia, Adriana Giret, and Vicente Botti

Abstract Recently a great number of methods and frameworks to develop multiagent systems have appeared. Nowadays there is no established framework to evaluate environments to develop multiagent systems (MAS) and choosing between one framework or another is a difficult task. The main contributions of this paper are: (1) a brief analysis of the state of the art in the evaluation of MAS engineering; (2) a complete list of criteria that helps in the evaluation of multiagent system development environments; (3) a quantitative evaluation technique; (4) an evaluation of the Ingenias methodology and its development environment using this evaluation framework.

1 INTRODUCTION

Nowadays, there is a great number of methods and frameworks to develop MAS, almost one for each agent-research group [11]. This situation makes the selection of one or another multiagent development tool, a very hard task. The main objective of this paper is to provide a mechanism to evaluate these kind of tools. This paper shows a list of criteria that allows a deep and complete analysis of multiagent development tools. Through this analysis, developers can evaluate the appropriateness of using a tool or another depending on their needs.

The rest of the paper is organized as follows: Section 1.1 briefly summarizes the state of the art of the evaluation of MAS engineering. Section 2 details some important features to develop MAS. Section 2 explains a quantitative technic to evaluate MASDKs (MultiAgent System Development Kits). In Section 3, the Ingenias methodology and it MASDK are presented. They are evaluated in Section **??**. Finally, Section 4 presents some conclusions and future works.

Emilia Garcia, Adriana Giret and Vicente Botti
Technical University of Valencia, e-mail: {mgarcia,agiret,vbotti}@dsic.upv.es

Please use the following format when citing this chapter:

Garcia, E., Giret, A. and Botti, V., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 3544.

1.1 Background

Shehory and Sturm [9] provide a list of criteria that includes software engineering related criteria and criteria relating to agent concepts. Also they add a metric evaluation. Cernuzzi and Rossi [3] present a qualitative evaluation criteria employing quantitative methods for the evaluation of agent-oriented analysis and design modeling methods. The related works focus their efforts on the analysis of methodologies, but do not analyzes the tools that provide support for these methodologies. It is a very important feature because a well-defined methodology loses a great part of its functionality if there is no tool to apply it easily.

Eiter and Mascardi [4] analyzes environments for developing software agents. They provide a methodology and general guidelines for selecting a MASDK. Their list of criteria includes agent features, software engineering support, agent and MAS implementation, technical issues of the MASDKs and finally economical aspects.

Bitting and Carter [2] use the criteria established by Eiter and Mascardi to analyze and compare five MASDKs. In order to obtain objective results from the evaluation Bitting and Carter add a quantitative evaluation. Sudeikat and Braunch [10] presents an interesting work in which they analyzes the gap between modeling and platform implementation. Their framework allows the evaluation of the appropriateness of methodologies with respect to platforms.

This paper is based on the related works. The main objective of this paper is to offer a list of evaluation criteria that allows to analyze and compare methods, techniques and environments for developing MAS. A metric is added to the qualitative criteria to allow a quantitative comparison. These criteria focus on the gap between the theorical guidelines of the methodologies and what can be modeled in the MASDKs. Furthermore, these criteria analyze the gap between the model and the final implementation, i.e., which implementation facilities provide the MASDKs and which model elements have no direct translation in the implementation platform.

2 CRITERIA

In this section, a list of evaluation criteria is described. These features allow a complete analysis of a MASDK and the selection between one and another. They are grouped in five categories.

2.1 Concepts and properties of MAS

As it is well known, there is no complete agreement on which features are mandatory to characterize an agent and a MAS. This is the reason why an analysis of the basic notions (concepts and properties) of agents and MAS are necessary at the beginning of the evaluation. This section deals with the question whether a methodology and it associated MASDK adhere to the basic concepts and properties of agents and MAS.

- AGENT FEATURES

These features are grouped into basic features that represent the core of agenthood, and advanced features that represent specific and desirable agent characteristics.

Basic features

Agent architecture: It represents the concepts that describe the internals of an agent. The importance of this feature is not to say which approach is better than other, but this feature is very useful to know if the approach is appropriate to specific requirements.

Properties: Agents are supposed to be autonomous, reactive, proactive and socials. In this section which agent properties are supported by the methodology and by the MASDK is analyzed.

Advanced features

Mental attitudes: The agent has mental notions like beliefs, desires, intentions and commitments.

Deliberative capabilities: The agent is able to select some possible plans to solve a problem and deliberate to choose the most appropriate in this situation.

Adaptivity: The adaptability feature may require that a modeling technique be modular and that it can activate each component according to the environmental state.

Meta-management: The agent is able to reason about a model of itself and of other agents.

- MAS FEATURES

Support for MAS organizations. At this point will be analyzed only which kind of organizations are supported, the other specific characteristics of the organizations will be analyzed in the following categories.

Support for the integration with services. Some MAS software engineering has been expanded to the integration with services [8]. At this point is interesting to analyze which kind of integration is supported by the approach (agents invoke services, services invoke agents or bidirectional) and the mechanisms used to facilitate the integration. Related with this, it is very interesting to know which services communication and specification standards are supported.

2.2 Software engineering support

The development of a MAS is a complex task that can be simplified with the use of MAS engineering techniques. This section will analyze how MASDKs support this techniques.

- APPLICATION DOMAIN

There are some methodologies and MASDKs thatcan be used to develop any kind of MAS, but other approaches are specialized in a particular application domain [5].

- MODEL-CENTRAL ELEMENT

Traditionally, agents are the model-central element in most MAS models, but in the last years there are an evolution to the organization-oriented modeling and service-oriented modeling.

- METHODOLOGY

Methodologies can be analyzed using the following criteria:

Based on metamodels. Meta-model presents relationships, entities, and diagrams, which are the elements to build MAS models.

Models dependence. A high dependence on some specific models of a modelling method may imply that if they are not well-designed it may affect all the design process; hence, lower dependence is better.

Development process. It indicates which software-development process follows the methodology.

Lifecycle coverage. In complex systems such as MAS it is desirable to use tools that facilitate the development of the application throughout the entire process.

Development guides. They facilitate the developers work and make the methodology more easy to understand and follow.

Platform dependent. Some methodologies are focused on the development in a specific deployment platform.

Organization support. The methodology includes agent-organization concepts in the development life cycle.

Service support. The methodology provides support to integrate services and agents at the different stages of the life cycle.

- MODELING LANGUAGE

The methodology should use a complete and unambiguous modeling language. It can be formal, informal or a mix of them. It should be expressive enough to represent MAS structure, data workflow, control workflow, communication protocols, concurrent activities and different abstraction level views. Other advanced features are the possibility to represent restrictions in the resources, mobil agents, the interaction with extern systems and the interaction with human beings.

- SUPPORT FOR ONTOLOGIES

Ontologies represent a powerful means to organize concepts and relations among concepts in an agent-based application, and to unambiguously describe features and properties of agents. At this point if the MASDK offers the possibility to model, implement or import ontologies is analyzed.

- VERIFICATION TOOLS

The verification process can be analyzed from two points of view:

Static verification. It involves to check the integrity of the system, i.e., that the specification of all model elements and the relationships between those elements are correct. The MASDK must be able to detect inconsistencies such as an agent who pursues a goal but the functionality of the agent does not allow it to achieve that goal. Furthermore, the MASDK notifies when the modeling is incomplete (for example, when there is an objective that is not achieve for anyone). In the best cases, the application not only detects these mistakes, but also proposes solutions.

Dynamic verification. It involves testing the system using simulations, i.e., the MASDK creates a simplified system prototype and test their behavior.

- THE GAP BETWEEN METHODS AND DEVELOPMENT TOOL

This section analyzes the gap between what is theoretically defined in the method-

ology and what can be modeled by the MASDK. Three conflicted areas have been highlighted.

Complete notation The MASDK should provide the possibility to model all the methodology elements and their relationships. All the restrictions defined in the methodology should be defined in the modeling language and should be taken into account in the MASDK.

Lifecycle coverage This criterion identifies which methodology stages are supported by the MASDK.

Development guidelines These guides are very useful to develop MAS and if they are integrated in the MASDK the development task become more intuitive and easier. This integration reduces the modeling time and facilitate the development of MAS to developers.

2.3 MAS implementation

This section analyzes how the MASDK helps the developer to transform the modeled system into a real application.

- IMPLEMENTATION FACILITIES

Graphical interfaces It represents the possibility to generate graphical interfaces using the MASDK.

Limited systems The MASDK may support the development of system with some limitations, i.e., the development of system that are going to be executed in limited devices like mobile phones.

Real time control Some application need real time control, so it must be supplied for the MASDK.

Security issues The MASDK can provide security mechanism to ensure that agents are not malicious and do not damage other agents, that their agents are not be damaged and has to avoid the interception or corruption of messages. These issues are more complex when the system has mobile agents or when agents interact with extern agents.

Physical environment models These are a library of simulators of physical parts of some kinds of systems for testing.

Code implementation The MASDK allows to implement or complete the agents code in the same tool.

Debugging facilities They are necessary for developing correct, reliable and robust system, due to the complex, distributed and concurrent nature of MAS.

- THE GAP BETWEEN MODELING AND IMPLEMENTATION

Match MAS abstractions with implementation elements

Agents use abstract concepts that are close to those used when reasoning, about human behaviours and organizations. This fact can facilitate the analysis and design activities but the gap between model and implementation increases.

Automatic code generation

It is an important feature because it reduces the implementation time and the errors in the code.

Code language This issue represents which programming language is used to generate agent code and which language is used to represent the ontologies.

Platform This issue represents for which agent platform is generated the code.

Generation technology Nowadays, there are a great number of techniques and languages to transform automatically from models to code.

Kind of generation It can be complete or it can generate only the skeletons of the agents. These skeletons usually have to be manually completed for the developer.

Utility agents There are different agents offering services that do not depend on the particular application domain (for example yellow and white pages). The MASDK should also provide them.

Reengineering These techniques are very useful in traditional software engineering and also in MAS development.

Services If the generating MAS is integrated with services, the MASDK should provide the necessary mechanisms for this integration.

2.4 Technical issues of the MASDK

This criteria selection is related to the technical characteristics of the development environment. Some of these features can have a dramatic impact on the usability and efficiency of this tools.

Programming language The language used to implement the MASDK and the language used to store the models are important keys.

Resources System requirements to the MASDK which include in which platforms can be executed and if it is light-weight.

Required expertise It indicates if it is necessary be a expert modeler and developer to use the MASDK.

Fast learning It indicates if the MASDK is easy to use and does not need much training time.

Possibility to interact with other applications For example this can provide the possibility to import or export models developed with other applications.

Extensible The MASDK is prepared to include other functional modules in an easy way.

Scalability This issue analyzes if the MASDK is ready to develop any scale of applications (small systems or large-scale applications).

Online help A desirable feature in a MASDK is that it helps developers when they are modeling or implementing, i.e., the MASDK takes part automatically or offer online suggestions to the developer.

Collaborative development This functionality may be very interesting to develop complex systems in which there are a group of developers which cooperates.

Documentation An important aspect when dealing with new proposals is how they are documented. A good documentation and technical support should be provided.

Examples If the MASDK presents complete case study is another feature to evaluate. The fact that the MASDK has been used in business environments also demonstrate the usefulness of the MASDK.

2.5 Economical aspects

Economical characteristics are important to choose between one or another MASDK. Obviously, one key in the evaluation is the cost of the application, the cost of it documentation and a technical service is provided. Also, the vendor organization gives an idea about the reliability and the continuity of the application.

2.6 Metric

A numerical evaluation offers a fast and general evaluation which allows to compare and evaluate methods and tools easily.

Each established criterion in Section 2 is associated with a weigh that represents the importance of this criterion. A ranking of 0 indicates that this criterion cannot be quantitative evaluated. For example, the use of one agent architecture or another cannot be evaluated as better or worst, it is only a feature and it will be more or less appropriate depending on the requirements of the system to develop. A ranking of 1 indicates that this criterion is desirable but not necessary. A ranking of 2 indicates that it is not necessary but very useful. A ranking of 3 indicates that it is necessary or very important in the MAS development. An evaluation vector for each MASDK is created stating how the approach covers each criterion. The scale of the points is 0, 25, 50, 75 or 100% depending on how feature is covered. The numerical evaluation is the result of the dot product between the weight vector and the evaluation vector.

The presented metric is based on [2] although this metric does the average of all the criteria without separating categories. In this paper the numerical evaluation is considered taking into account the categories established in Section 2 to detect which parts of the MASDKs has more lacks and should be improved.

	Concepts and properties of agents and MAS										
	Agent basic features					Agent advanced features				MAS features	
	Agent architecture	Propierties				Mental attitudes	Deliberative	Meta-management	Adaptability	Organizations	Services
		Autonomy	Reactivity	Proactivity	Sociability						
Weight	0	3	3	3	3	3	2	1	1	2	1
Qualitative ev.	BDI	Yes	No	Yes	Yes	Yes	Yes	No	No	Yes	No
Numeric ev.	----	●	●	●	●	●	◕	○	○	◕	○

Note: ○ is 0% ◔ is 0-25% ◒ is 25-50% ◕ is 50-75% ● is 75-100%

Fig. 1 Concepts and properties of MAS evaluation.

	Softare engineering support																		
	Domain	Central element	Methodology							Modeling language			Ontologies	Verification		Gap Methods-Model tool		
			Meta models	Models´ dependence	Development process		Platform dependent	Organization support	Service support	Type	Expressiveness			Static	Dynamic	Notation	Lifecycle coverage	Guidelines
					Lifecycle coverage	Guidelines					Basic	Advanced						
Weight	---	---	3	2	3	2	2	2	1	---	3	2	3	3	2	3	3	2
Qualitative ev.	General	Organizations	Yes	High	Analysis, Design, some of Implementation	Yes	Independent	Yes	No	Informal	Yes	No	No	Yes	No	No gap	Yes	No
Numeric ev.	---	---	●	◒	◕	◕	●	◕	○	---	●	○	○	◕	○	●	●	○

Fig. 2 Software engineering support evaluation.

3 CASE STUDY: INGENIAS

INGENIAS [7] is a methodology for the development of MAS that is supported by an integrated set of tools, the INGENIAS Development Kit (IDK) [6]. These tools include an editor to create and modify MAS models, and a set of modules for code generation and verification of properties from these models. In this section Ingenias and the IDK are presented and evaluated according to the framework and the metric presented in Section 2.

- CONCEPTS AND PROPERTIES OF MAS

Ingenias agents follow a BDI architecture and have all the basic properties defined by Wooldridge. Also Ingenias specifies mental attitudes and deliberative capabilities but it does not provide support to specify adaptivity, meta-management or emotionality (see Figure 1). Ingenias specifies an organizational model in which groups, members, workflows and organizational goals are described. Ingenias does not model social norms or the dynamic of the organization, i.e., how an agent can enter in a group, etc. At this moment, Ingenias does not support service integration.

- SOFTWARE ENGINEERING SUPPORT

Ingenias is a general application domain and its model-central element are organizations (Figure 2). It integrates results from research in the area of agent technology with a well-established software development process, which in this case is the Rational Unified Process (RUP). This methodology defines five meta-models that describe the elements that form a MAS from several viewpoints, and that allow to define a specification language for MAS. These metamodels are: Organization meta-model, Environment meta-model, Tasks/Goals meta-model, Agent meta-model, Interaction meta-model. These metamodels have strong dependences and relations. Ingenias provides a development guides for the analysis and design stages. These guides include organization support and have no platform dependences. Furthermore, Ingenias provide some support for the implementation stage. Ingenias also provides an informal modeling language that provides mechanism to define all the basic features explained in Section 2 but that it is not useful to represent mobil agents and the other advanced-related features. The IDK offers a module that realize a static verification of the modeled system. It is very useful because it helps developers to find errors in their models and suggests possible solutions. As is shown in Figure 5, Ingenias fills well the gap between its methodology and the development tool. The elements of the methodology and the notation of the modeling language is totally supported by the IDK. Also all the stage that are covered by the methodology, are covered by the IDK as well. At this moment the IDK has no development guidelines integrated, but there are some researches about this topic.

- MAS IMPLEMENTATION

Figure 5 shows that Ingenias does not provide a good support to the implementation stage. It has a module to transform models into Jade agents [1]. This module generate skeletons of the agents and their tasks. Almost all the elements of the methodology can be directly transformed into elements of the Jade platform, but there are some abstraction like goals or mental states that have not a corresponding element of the platform.

	MAS implementation															
	Implementation facilities							Gap modeling-implementation								
									Automatic code generation							
	Interfaces	Limited systems	Real Time	Security	Physical environments	Code impleme ntation	Debugging	Matching	Code language	Platform	Technology	Kind	Utility	Reengineery	Services	
Weight	1	1	1	1	1	1	1	3	---	---	---	2	1	1	1	
Qualitative ev.	No	No	No	No	No	No	No	Yes	Java	Jade	Templates	Skeletons	No	No	No	
Numeric ev.	○	○	○	○	○	○	○	◔	---	---	---	◒	○	○	○	

Fig. 3 MAS implementation issues of the MASDK evaluation.

	Technical issues of the MASDK										Economical aspects			
	Programming language	Requirements	Required expertise	Fast learning	Interactions	Extensible	Online help	Collaborative	Documentation	Examples	Vendor	Cost	Updates	Technical service
Weight	0	1	2	2	1	2	2	1	3	3	---	2	3	3
Qualitative ev.	Java / Xml	Multiplatform	Yes / not very intuitive	Medium	no	Yes	No	No	Yes	Yes	Grasia	Free	Yes	Yes
Numeric ev.	-----	●	◒	◒	○	●	○	○	●	◔	---	●	●	◔

Fig. 4 Technical issues of the MASDK evaluation.

Metric evaluation		Score (percent)		Comentaries
Concepts and properties of agents and MAS	Agent basic features(3)	100,00	82,86	Obtains a good result because the Ingenias methodology supports all the basic concepts and features of agents and also offers support to organizations.
	Agent advanced features(2)	90,00		
	MAS features(2)	50,00		
Software engineering support	Methodology(3)	71,67	53,93	Ingenias provides a well-defined methodology which is well-supported by its modeling language. Furthermore, it obtain a good result in the category "Gap methods-tools", so this methods are well-supported by its IDK and almost there is no gap between what is defined by the methodology and what is represented with the MASDK.
	Modeling language(3)	60,00		
	Ontology(2)	0,00		
	Verification(3)	45,00		
	Gap methods-tool(3)	75,00		
MAS implementation	Implementation facilities(3)	22,50	17,50	The MAS implementation is an ongoing work topic for the Ingenias developers. At this moment, it is not well implemented by the IDK and there are many issues that there are not covered.
	Gap model-implementation(2)	10,00		
Technical issues of the MASDK		60,29		The IDK is technically well-defined, but there are some interesting features that are not covered.
Economical aspects		90,63		The result is very good, the reason is that Ingenias is free and is well-supported by its authors.

Fig. 5 Numerical evaluation results.

- TECHNICAL ISSUES OF THE MASDK

The IDK has been implemented in Java, because of that it is multiplatfom. A new functionality module can be added to the IDK easily. The IDK is not very intuitive, but it does not require much time to learn it. The documentation of Ingenias and of its IDK is complete and some validated examples are provided with the IDK. Despite this, the examples are not completely developed and some of them have modeling mistakes.

- ECONOMICAL ASPECTS

Ingenias is an academical work developed in the Grasia! research group. This project is still open, so they are offering new versions of the IDK and improving the methodology. Ingenias and the IDK are open source and all their documentation is publicly available. There is no specific technical service but authors answer questions by email and by the gap repository.

- NUMERICAL EVALUATION

Figure 5 shows the numerical evaluation results. The final result of each category is the dot product between the weight of the features analyzed (the number inside the parentheses of the second column) and their numerical result (third column). Developers can obtain a fast overview about Ingenias and its IDK looking this figure.

4 CONCLUSION

This paper summarizes the state of the art in the evaluation of methods and tools to develop MAS. These studies are the base of the presented evaluation framework. This framework helps to evaluate MASDKs by the definition of a list of criteria that allows to analyze the main features of this kind of systems. This list covers traditional software engineering needs and specific characteristics for developing MAS. This study helps in the evaluation of the gap between the methods and the modeling tool, and the gap between the model and the implementation.

A quantitative evaluation method is presented. It allows a numeric evaluation and comparison. It gives developers a fast overview of the quality of the evaluated MASDK in each category. The weight of the criteria is a variable feature that affect the result of the evaluation. This feature provides developers a mechanism to adapt the evaluation framework to their own requirements.

This evaluating framework has been used successfully to evaluate Ingenias and its IDK. The results of the evaluation shows that this approach covers the entire development process in a basic way, but, it has important lacks in the transformation from models to the final implementation. It should improve it implementation coverage. As future work, this framework will be used to evaluate and compare a large set of MASDKs and their methodologies.

Acknowledgements This work is partially supported by the TIN2006-14630-C03-01, PAID-06-07/3191 projects and CONSOLIDER-INGENIO 2010 under grant CSD2007-00022.

References

1. F. L. Bellifemine, G. Caire, and D. Greenwood. *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. John Wiley & Sons, 2007.
2. E. Bitting, J. Carter, and A. A. Ghorbani. Multiagent System Development Kits: An Evaluation. In *Proc. of the CNSR*, pages 80–92, 2003.
3. L. Cernuzzi and G. Rossi. On the evaluation of agent oriented modeling methods. In *In Proceedings of Agent Oriented Methodology Workshop*, 2002.
4. T. Eiter and V. Mascardi. Comparing environments for developing software agents. *AI Commun.*, 15(4):169–197, 2002.
5. A. Giret, V. Botti, and S. Valero. *MAS Methodology for HMS*, volume LNAI 3593, pages 39–49. Springer Verlag., 2005.
6. J. Gomez-Sanz and J. Pavon. Ingenias development kit (idk) manual, version 2.5. 2005.
7. J. Pavon, J. Gomez-Sanz, and R. Fuentes. *The INGENIAS Methodology and Tools*, volume chapter IX, page 236276. Henderson-Sellers, 2005.
8. M. P. Singh and M. N. Huhns. *Service-Oriented Computing Semantics, Processes, Agents*. John Wisley and Sons Ltd, 2005.
9. A. Sturm and O. Shehory. A framework for evaluating agent-oriented methodologies. In *AOIS*, volume 3030 of *LNCS*, pages 94–109. Springer, 2003.
10. J. Sudeikat, L. Braubach, A. Pokahr, and W. Lamersdorf. Evaluation of agent-oriented software methodologies examination of the gap between modeling and platform. *AOSE-2004 at AAMAS04*, 2004.
11. M. Wooldridge and P. Ciancarini. Agent-Oriented Software Engineering: The State of the Art. In *AOSE01*, volume 1957/2001 of *LNCS*, pages 55–82. Springer, 2001.

Information-Based Planning and Strategies

John Debenham

Abstract The foundations of information-based agency are described, and the principal architectural components are introduced. The agent's deliberative planning mechanism manages interaction using plans and strategies in the context of the relationships the agent has with other agents, and is the means by which those relationships develop. Finally strategies are described that employ the deliberative mechanism and manage argumentative dialogues with the aim of achieving the agent's goals.

1 Introduction

This paper is in the area labelled: *information-based agency* [9]. Information-based agency is founded on two premises. First, everything in its world model is uncertain [2]. Second, everything that an agent communicates gives away valuable information. Information, including arguments, may have no particular utilitarian value [6], and so may not readily be accommodated by an agent's utilitarian machinery.

An information-based agent has an identity, values, needs, plans and strategies all of which are expressed using a fixed ontology in probabilistic logic for internal representation and in an illocutionary language [8] for communication. All of the forgoing is represented in the agent's deliberative machinery. We assume that such an agent resides in a electronic institution [1] and is aware of the prevailing norms and interaction protocols. In line with our "Information Principle" [8], an information-based agent makes no *a priori* assumptions about the states of the world or the other agents in it — these are represented in a world model, $\mathscr{M}^t$, that is inferred solely from the messages that it receives.

The world model, $\mathscr{M}^t$, is a set of probability distributions for a set of random variables each of which represents the agent's expectations about some point of

John Debenham
University of Technology, Sydney, Australia e-mail: debenham@it.uts.edu.au

Please use the following format when citing this chapter:

Debenham, J., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 45–54.

interest about the world or the other agents in it. We build a history of interaction by noting each commitment made (commitments to act, commitments to the truth of information or to the validity of an argument), and by relating each of them to subsequent observations of what occurs. Tools from information theory are then used to summarise these historic (commitment, observation) pairs — in this way we have defined models of *trust*, *honour*, *reliability* and *reputation* [8]. Further we have defined the *intimacy* and *balance* of both dialogues and relationships [10] in terms of our 'LOGIC' illocutionary framework. All of these notions make no presumption that our agents will align themselves with any particular strategy.

In related papers we have focussed on argumentation strategies, trust and honour, and have simply assumed that the agent has a kernel deliberative system. In this paper we describe the deliberative system for an information-base agent.

2 Plans

A *plan* p is $p(a_p, s_p, t_p, u_p, c_p, g_p)$ where:

- a_p is a conditional action sequence — i.e. it is conditional on future states of the world, and on the future actions of other agents. We think of plans as probabilistic statecharts in the normal way where the arcs from a state are labelled with "event / condition / action" leading into a P symbol that represents the lottery, s_p, that determines the next state as described following:
- $s_p : S \to \mathbb{P}(S_p = s) \equiv \mathbf{s}$ where S is the set of states and S_p is a random variable denoting the state of the world when a_p terminates[1].
- $t_p : S \to \mathbb{P}(T_p = t) \equiv \mathbf{t}$ where T_p is a random variable denoting the time that a_p takes to execute and terminate for some finite set of positive time interval values for t.
- $u_p : S \to \mathbb{P}(U_p = u) \equiv \mathbf{u}$ where U_p is a random variable denoting the gross utility gain, excluding the cost of the execution of a_p for some finite set of utility values for u.
- $c_p : S \to \mathbb{P}(C_p = c) \equiv \mathbf{c}$ where C_p is a random variable denoting the cost of the execution of a_p for some finite set of cost values for c.
- $g_p : S \to \mathbb{P}(G_p = g) \equiv \mathbf{g}$ where G_p is a random variable denoting the expected information gain to α *and* to β of the dialogue that takes place during the execution of the plan each expressed in $\mathscr{G} = \mathscr{F} \times \mathscr{O}$.

The distributions above are estimated by observing the performance of the plans as we now describe.[2] In the absence of any observations the probability mass functions for S_p, T_p, U_p, C_p and G_p all decay at each and every time step by:

[1] For convenience we assume that all action sequences have a "time out" and so will halt after some finite time.

[2] An obvious simplification would be to use point estimates for t_p, u_p, c_p and each element of g_p, but that is too weak a model to enable comparison.

$$\mathbb{P}^{t+1}(X_i) = \lambda \times \mathbb{D}(X_i) + (1-\lambda) \times \mathbb{P}^t(X_i) \tag{1}$$

for some constant $\lambda : 0 < \lambda < 1$, where λ is the *decay rate*.

The implementation of a_p does not concern us. We do assume that the way in which the plans are implemented enables the identification of common algorithms and maybe common methods within different plans. Given two plans p and q, the function $\mathrm{Sim}(p,q) \in [0,1]$ measures the similarity of their action sequences a_p and a_q in the sense that their performance parameters are expected to be correlated to some degree.

Estimating S_p. Denote the prior estimate by $\mathbf{s^t}$. When a plan terminates, or is terminated, the world will be in one of p's end states. Call that state z. Then the observed distribution for $\mathbf{s^{t+\delta t}}$ will have the value 1 in position z. On the basis of this observation the agent may be inclined to fix its estimate for s_z^{t+1} at γ where $s_z^t \leq \gamma \leq 1$. The posterior distribution $\mathbf{s^{t+1}}$ is defined as the distribution with minimum relative entropy with respect to $\mathbf{s^t}$: $\mathbf{s^{t+1}} = \arg\min_{\mathbf{r}} \sum_j r_j \log \frac{r_j}{\mathbf{s^t}}$ that satisfies the constraint $s_z^{t+1} = \gamma$. If $\gamma = s_z^t$ then the posterior is the same as the prior. If $\gamma = 1$ then the posterior is certain with $\mathbb{H}(\mathbf{s^{t+1}}) = 0$. One neat way to calibrate γ is in terms of the resulting information gain; that is to measure γ in terms of the resulting *learning rate* μ:

$$\mathbb{H}(\mathbf{s^{t+1}}) = (1-\mu) \times \mathbb{H}(\mathbf{s^t}) \tag{2}$$

where $\mu : 0 < \mu < 1$.

Estimating T_p, U_p, C_p and G_p. Just as for estimating S_p, when the plan terminates α will have observations for the values of these variables, and as a result may wish to increase the corresponding frequency in the posterior to some new value. Using the method described above for estimating S_p, the posterior distribution is the distribution with minimum relative entropy with respect to the prior subject to the constraint that the frequency corresponding to the observation is increased accordingly.

Further, for these four variables we use the $\mathrm{Sim}(\cdot,\cdot)$ function to revise the estimates for 'nearby' plans. In [9] two methods for using a $\mathrm{Sim}(\cdot,\cdot)$ function to revise estimates are described — the situation here is rather simpler. Consider the variable C_p. Applying the method in the paragraph 'Estimating S_p.', suppose a value had been observed for C_p and as a result of which c_j^{t+1} had been constrained to be γ. Consider any plan q for which $\mathrm{Sim}(p,q) > 0$. Denote $\mathbb{P}(C_q = c)$ by $\mathbf{d}$. The posterior distribution $\mathbf{d^{t+1}}$ is defined as the distribution with minimum relative entropy with respect to $\mathbf{d^t}$: $\mathbf{d^{t+1}} = \arg\min_{\mathbf{r}} \sum_j r_j \log \frac{r_j}{\mathbf{d^t}}$ that satisfies the constraint: $d_j^{t+1} = \gamma'$ where γ' is such that:

$$\mathbb{H}(\mathbf{d^{t+1}}) = (1-\mu \times \mathrm{Sim}(p,q)) \times \mathbb{H}(\mathbf{d^t}) \tag{3}$$

where $0 \leq \mathrm{Sim}(p,q) \leq 1$ with higher values indicating greater similarity.

3 Planning

If an agent's needs could potentially be satisfied by more than one plan then a mechanism is required to select which plan to use. As the execution of plans incurs a cost we assume that α won't simply fire off every plan that may prove to be useful. A random variable, V_p, derived from the expectations of S_p, T_p, U_p, C_p, G_p and other estimates in $\mathscr{M}^t$ represents the agent's expectations of each plan's overall *performance*. V_p is expressed over some finite, numerical valuation space with higher values being preferred.

The mechanisms that we describe all operate by selecting plans stochastically. We assume that there is a set of P candidate plans $\{p_i\}$ with corresponding random variables V_{p_i} representing performance, and plan p_j is chosen with probability q_j where $\sum_k q_k = 1$. Let $\mathscr{N}^t = \{V^t_{p_k}\}_{k=1}^P$. The integrity of the performance estimates for random variable V_{p_i} are maintained using the method "Estimating S_p" in Section 2. If p_i is selected at time t then when it terminates the observed performance, $v^t_{p_i,\text{ob}}$, is fed into that method.

First, consider the naïve mechanism that selects plan p_j by: $q_j = 1$ for $j = \arg\max_i \mathbb{E}(V_{p_i})$. This mechanism is well-suited to a one-off situation. But if the agent has continuing need of a set of plans then choosing the plan with highest expected payoff may mean that some plans will not be selected for a while by which time their performance estimates will have decayed by Equation 1 to such a extent that may never be chosen. An agent faces the following dilemma: the only way to preserve a reasonably accurate estimate of plans is to select them sufficiently often — even if they they don't perform well today perhaps one day they will shine.

The simple method: $q_i = \frac{1}{P}$ selects all plans with equal probability. The following method attempts to prevent the uncertainty of estimates from decaying above a threshold, τ, by setting $q_j = 1$ where:

if $\exists i \cdot \mathbb{H}(V_{p_i}) > \tau$ **then** let $j = \arg\max_k \mathbb{H}(V_{p_k})$

else let $j = \arg\max_k \mathbb{E}(V_{p_k})$

this method may deliver poor performance from the '**then**' and good performance from the '**else**', but at least it attempts to maintain some level of integrity of the performance estimates, even if it does so in an elementary way.

A strategy is reported in [4] on how to place all of one's wealth as win-bets indefinitely on successive horse races so as to maximise the rate of growth; this is achieved by proportional gambling, i.e. by betting a proportion of one's wealth on each horse equal to the probability that that horse will win. This result is interesting as the strategy is independent of the betting odds. Whether it will make money will depend on the punter's ability to estimate the probabilities better than the bookmaker. The situation that we have is not equivalent to the horse race, but it is tempting to suggest the strategies:

$$q_i = \frac{\mathbb{E}(V_{p_i})}{\sum_k \mathbb{E}(V_{p_k})} \tag{4}$$

$$q_i = \mathbb{P}(V_{p_i} > V_{p_j}), \forall V_{p_j} \in \mathscr{N}, j \neq i \tag{5}$$

For the second strategy: q_i is the probability that p_i's performance is the better than that of all the other plans. With this definition it is clear that $\sum_i q_i = 1$. Both strategies will favour those plans with a better performance history. Whether they will prevent the integrity of the estimates for plans with a poor history from decaying to a meaningless level will depend on the value of λ in Equation 1, the value of μ in Equation 2, and on the frequency with which plans are activated. As the estimates for plans that perform well, and plans that perform badly, all decay to the maximum entropy decay limit $\mathbb{D}(V_{p_i})$ if they are not invoked, both of these strategies indirectly take account of the level of certainty in the various performance estimates.

We consider now the stability of the integrity of the performance estimates in time. If plan p_j is *not* executed the information loss in X_j^t for one time step due to the effect of Equation 1 is: $\lambda \times \mathbb{H}(X_j^t)$. If no plans in $\mathscr{N}$ are executed during one time step then the total information loss in $\mathscr{N}$ is: $\lambda \times \sum_k \mathbb{H}(X_k^t)$. If plan p_j *is* executed the information gain in X_j^t due to the effect of Equation 2 is: $\mu \times \mathbb{H}(X_j^t)$, but this observation may effect the other variables in $\mathscr{N}^t$ due to Equation 3, and the total information gain in $\mathscr{N}$ is: $\mu \times \sum_k \mathrm{Sim}(p_j, p_k) \times \mathbb{H}(X_k^t)$. Assuming that at most one plan in $\mathscr{N}^t$ is executed during any time step, and that the probability of one plan being executed in any time step is χ; the expected net information gain of $\mathscr{N}^{t+1}$ compared with $\mathscr{N}^t$ is:

$$\chi \cdot \mu \cdot \sum_j q_j \cdot \sum_k \mathrm{Sim}(p_j, p_k) \cdot \mathbb{H}(X_k^t) - \lambda \cdot \sum_k \mathbb{H}(X_k^t) \tag{6}$$

If this quantity is negative then the agent may decide to take additional steps to gain performance measurements so as to avoid the integrity of these estimates from consistently declining.

We now consider the parameters λ and μ to be used with the strategy in Equation 4. The effect of Equation 1 on variable V_i after t units of time is:

$$\left(1 - (1-\lambda)^t\right) \times \mathbb{D}(V_{p_i}) + (1-\lambda)^t \times V_{p_i}^{t_0}$$

The probability that plan p_i will be activated at any particular time is:

$$\chi \times \frac{\mathbb{E}(V_{p_i})}{\sum_k \mathbb{E}(V_{p_k})}$$

and the mean of these probabilities for all plans is: $\frac{\chi}{P}$. So the mean number of time units between each plan's activation is: $\frac{N}{\chi}$. In the absence of any intuitive value for λ, a convenient way to calibrate λ is in terms of the expected total decay towards $\mathbb{D}(V_{p_i})$ between each activation — this is expressed as some constant ϕ, where $0 < \phi < 1$. For example, $\phi = \frac{1}{2}$ means that we expect a 50% decay between activations. The value of λ that will achieve this is: $\lambda = 1 - (1-\phi)^{\chi \div N}$. Then the value for μ is chosen so that the expression (6) is non-negative. Using these values should ensure that the probability distributions for the random variables V_i remain within reasonable bounds, and so remain reasonably discriminating.

It would be nice to derive a method that was optimal in some sense, but this is unrealistic if the only data available is historic data such as the V_{p_i}. In real situations the past may predict the future to some degree, but can not be expected to predict performance outcomes that are a result of interactions with other autonomous agents in a changing environment. As a compromise, we propose to use (5) with values for λ and μ determined as above. (5) works with the whole distribution rather than (4) that works only with point estimates, but is algebraically simpler. These methods are proposed on the basis that the historic observations are all that α has.

4 Preferences

Agent α's *preferences* is a relation defined over an *outcome space*, where $s_1 \prec_\alpha s_2$ denotes "α prefers s_2 to s_1". Elements in the outcome space may be described either by the world being in a certain state or by a concept in the ontology having a certain value. If an agent knows its preferences then it may use results from game theory or decision theory to achieve a preferred outcome in some sense. For example, an agent may prefer the concept of price (from the ontology) to have lower values than higher, or to purchase wine when it is advertised at a discount (a world state). In practice the articulation of a preference relation may not be simple.

Consider the problem of specifying a preference relation for a collection of fifty cameras with different features, from different makers, with different prices, both new and second hand. This is a multi-issue evaluation problem. It is realistic to suggest that "a normal intelligent human being" may not be able to place the fifty cameras in a preference ordering with certainty, or even to construct a meaningful probability distribution to describe it. The complexity of articulating preferences over real negotiation spaces poses a practical limitation on the application of preference-based strategies.

In contract negotiation the outcome of the negotiation, (a', b'), is the enactment of the commitments, (a, b), in that contract, where a is α's commitment and b is β's. Some of the great disasters in market design [5], for example the Australian Foxtel fiasco, could have been avoided if the designers had considered how the agents were expected to deviate (a', b') from their commitments (a, b) after the contract is signed.

Consider a contract (a, b), and let $(\mathbb{P}^t_\alpha(a'|a), \mathbb{P}^t_\alpha(b'|b))$ denote α's estimate of what will be enacted if (a, b) is signed. Further assume that the pair of distributions $\mathbb{P}^t_\alpha(a'|a)$ and $\mathbb{P}^t_\alpha(b'|b)$ are independent [3][3] and that α is able to estimate $\mathbb{P}^t_\alpha(a'|a)$ with confidence. α will only be confident in her estimate of $\mathbb{P}^t_\alpha(b'|b)$ if β's actions are constrained by norms, or if α has established a high degree of trust in β. If α is unable to estimate $\mathbb{P}^t_\alpha(b'|b)$ with reasonable certainty then put simply: she won't know what she is signing. For a utilitarian α, $(a_1, b_1) \prec_\alpha (a_2, b_2)$ if she prefers $(\mathbb{P}^t_\alpha(a'_2|a_2), \mathbb{P}^t_\alpha(b'_2|b_2))$ to $(\mathbb{P}^t_\alpha(a'_1|a_1), \mathbb{P}^t_\alpha(b'_1|b_1))$ in some sense.

[3] That is we assume that while α is executing commitment a she is oblivious to how β is executing commitment b and *vice versa*.

One way to manage contract acceptance when the agent's preferences are unknown is to found the acceptance criterion instead on the simpler question: "how certain am I that (a,b) is a good contract to sign?" — under realistic conditions this is easy to estimate[4].

So far we have not considered the management of information exchange. When a negotiation terminates it is normal for agents to review what the negotiation has cost *ex post*; for example, "I got him to sign up, but had to tell him about our plans to close our office in Girona". It is not feasible to attach an intrinsic value to information that is related to the value derived from enactments. Without knowing what use the recipient will make of the "Girona information", it is not possible to relate the value of this act of information revelation to outcomes and so to preferences.

While this negotiation is taking place how is the agent to decide whether to reveal the "Girona information"? He won't know then whether the negotiation will terminate with a signed contract, or what use the recipient may be able to make of the information in future, or how any such use might affect him. In general it is unfeasible to form an expectation over these things. So we argue that the decision of whether to reveal a piece of information should *not* be founded on anticipated negotiation outcomes, and so this decision should not be seen in relation to the agent's preferences. The difficulty here is that value is derived from information in a fundamentally different way to the realisation of value from owning a commodity, for example[5].

A preference-based strategy may call upon powerful ideas from game theory. For example, to consider equilibria α will require estimates of $\mathbb{P}^t_\beta(a'|a)$ and $\mathbb{P}^t_\beta(b'|b)$ in addition to $\mathbb{P}^t_\alpha(a'|a)$ and $\mathbb{P}^t_\alpha(b'|b)$ — these estimates may well be even more speculative than those in the previous paragraph. In addition she will require knowledge about β's utility function. In simple situations this information may be known, but in general it will not.

5 Information-based strategies

An information-based agent's deliberative logic consists of:

1. The agent's *raison d'être* — its mission — this may not be represented in the agent's code, and may be implicit in the agent's design.

[4] In multi-issue negotiation an agent's preferences over each individual issue may be known with certainty. Eg: she may prefer to pay less than pay more, she may prefer to have some feature to not having it. In such a case, if some deals are known to be unacceptable with certainty, some are known to be acceptable with certainty, and, perhaps some known to be acceptable to some degree of certainty then maximum entropy logic may be applied to construct a complete distribution representing 'certainty of acceptability' over the complete deal space. This unique distribution will be *consistent* with what *is* known, and *maximally noncommittal* with respect to what is *not* known.

[5] If a dialogue is not concerned with the exchange of anything with utilitarian value, then the two agents may feel comfortable to balance the information exchanged using the methods in [10].

2. A set of *values*, Π, — high level principles — and a fuzzy function $\upsilon : (S \times A \times \Pi) \rightarrow$ fuz, that estimates, when the world is in state $s \in S$, whether the agent performing action $a \in A$ supports or violates a value $\pi \in \Pi$.
3. A *strategy* that provides an overarching context within which the plans are executed — see Section 5. The strategy is responsible for the evolution of the relationships between the agents, and for ensuring that plans take account of the state of those relationships.
4. A hierarchy[6] of *needs*, N, and a function $\sigma : N \rightarrow \mathscr{P}(S)$ were $\sigma(n)$ is the set of states that satisfy need $n \in N$. Needs turn 'on' spontaneously, and in response to *triggers*, T; they turn 'off' because the agent believes they are satisfied.
5. A set of *plans*, P — Section 2.

In this model an agent knows with certainty those states that will satisfy a need, but does *not* know with certainty what state the world is in.

We now describe the strategic reasoning of an information-based agent. This takes account of the, sometimes conflicting, utilitarian and information measures of utterances in dialogues and relationships. This general definition may be instantiated by specifying functions for the ψ_i in the following.

The following notation is used below. R_i^t denotes the relationship (i.e. the set of all dialogues) between α and β_i at time t. *Intimacy* is a summary measure of a relationship or a dialogue and is represented in $\mathscr{G}$. We write I_i^t to denote the intimacy of that relationship, and $I(d)$ to denote the intimacy of dialogue d. Likewise B_i^t and $B(d)$ denotes balance.

The Needs Model. α is driven by its needs. When a need fires, a plan is chosen to satisfy that need using the method in Section 3. If α is to contemplate the future she will need some idea of her future needs — this is represented in her *needs model*: $\nu : \mathscr{T} \rightarrow \times^n[0,1]$ where $\mathscr{T}$ is time, and: $\nu(t) = (n_1^t, \ldots, n_N^t)$ where $n_i^t = \mathbb{P}(\text{need } i \text{ fires at time } t)$.

Setting Relationship Targets. On completion of each dialogue of which α is a part, she revises her aspirations concerning her intimacy with all the other agents. These aspirations are represented as a *relationship target*, T_i^t, for each β_i, that is represented in $\mathscr{G}$. Let $\mathbf{I}^t = (I_1^t, \ldots, I_o^t)$, $\mathbf{B}^t = (B_1^t, \ldots, B_o^t)$ and $\mathbf{T}^t = (T_1^t, \ldots, T_o^t)$, then $\mathbf{T}^t = \psi_1(\nu, \mathbf{I}^t, \mathbf{B}^t)$ — this function takes account of all β_i and aims to encapsulate an answer to the question: "Given the state of my relationships with my trading partners, what is a realistic set of relationships to aim for in satisfaction of my needs?".

Activating Plans. If at time t, some of α's active needs, N_{active}^t, are not adequately[7] being catered for, N_{neglect}^t, by existing active plans, P_{active}^t, then select P_{active}^{t+1} to take account of those needs:

---

[6] In the sense of the well-known Maslow hierarchy [7], where the satisfaction of needs that are lower in the hierarchy take precedence over the satisfaction of needs that are higher.

[7] For each need n, $\sigma(n)$ is the set of states that will satisfy n. For each active plan p, $\mathbb{P}(S_p = s)$ is probability distribution over the possible terminal states for p. During p's execution this initial estimation of the terminal state is revised by taking account of the known terminal states of executed sub-plans and $\mathbb{P}(S_{p'} = s)$ for currently active sub-plans p' chosen by p to satisfy subgoals. In this way we continually revise the probability that P_{active}^{t-1} will satisfy α's active needs.

$$P^{t+1}_{\text{active}} = \psi_2(P^t_{\text{active}}, N^t_{\text{neglect}}, N^t_{\text{active}}, \mathbf{I}^t, \mathbf{T}^t)$$

The idea being that α will wish select P^{t+1}_{active} so as to move each observed intimacy I^t_i towards its relationship target intimacy T^t_i. Having selected a plan p, $\mathbb{E}(U_p)$ and $\mathbb{E}(G_p)$ assist α to set the *dialogue target*, D^t_i, for the current dialogue [10]. In Section 3 we based the plan selection process on a random variable V_p that estimates the plan's performance in some way. If α is preference-aware then V_p may be defined in terms of its preferences.

Deactivating Plans. If at time t, a subset of α's active plans, $P^t_{\text{sub}} \subset P^t_{\text{active}}$, adequately caters for α's active needs, N^t_{active}, then:

$$P^{t+1}_{\text{active}} = \psi_3(P^t_{\text{active}}, N^t_{\text{active}}, \mathbf{I}^t, \mathbf{T}^t)$$

is a minimal set of plans that adequately cater for N^t_{active} in the sense described above. The idea here is that P^{t+1}_{active} will be chosen to best move the observed intimacy I^t_i towards the relationship target intimacy T^t_i as in the previous paragraph.

The work so far describes the selection of plans. Once selected a plan will determine the actions that α makes where an action is to transmit an utterance to some agent determined by that plan. Plans may be bound by interaction protocols specified by the host institution.

Executing a Plan — Options. [10] distinguishes between a *strategy* that determines an agent's Options from which a single kernel action, a, is selected; and *tactics* that wrap that action in argumentation, a^+ — that distinction is retained below. Suppose that α has adopted plan p that aims to satisfy need n, and that a dialogue d has commenced, and that α wishes to transmit some utterance, u, to some agent β_i. In a multi-issue negotiation, a plan p will, in general, determine a set of Options, $A^t_p(d)$ — if α is preference aware [Section 4] then this set could be chosen so that these options have similar utility. Select a from $A^t_p(d)$ by:

$$a = \psi_4(A^t_p(d), \Pi, D^t_i, I(d), B(d))$$

that is the action selected from $A^t_p(d)$ will be determined by α's set of values, Π, and the contribution a makes to the development of intimacy.

If d is a bilateral, multi-issue negotiation we note four ways that information may be used to select a from $A^t_p(d)$. (1) α may select a so that it gives β_i similar information gain as β_i's previous utterance gave to α. (2) If a is to be the opening utterance in d then α should avoid making excessive information revelation due to ignorance of β_i's position and should say as little as possible. (3) If a requires some response (e.g. a may be an offer for β_i to accept or reject) then α may select a to give her greatest expected information gain about β_i's private information from that response, where the information gain is either measured overall or restricted to some area of interest in $\mathscr{M}^t$. (4) If a is in response to an utterance a' from β_i (such as an offer) then α may use entropy-based inference to estimate the probability that she should accept the terms in a' using nearby offers for which she knows their acceptability with certainty [9].

Executing a Plan — Tactics. The previous paragraph determined a kernel action, a. Tactics are concerned with wrapping that kernel action in argumentation, a^+. To achieve this we look beyond the current action to the role that the dialogue plays in the development of the relationship:

$$a^+ = \psi_5(a, , T_i^t, I_i^t, I(d), B_i^t, B(d))$$

In [10] *stance* is meant as random noise applied to the action sequence to prevent other agent's from decrypting α's plans. Stance is important to the argumentation process but is not discussed here.

6 Conclusion

In this paper we have presented a number of measures to value information including a new model of confidentiality. We have introduced a planning framework based on the kernel components of an information-based agent architecture (i.e. decay, semantic similarity, entropy and expectations). We have defined the notion of strategy as a control level over the needs, values, plans and world model of an agent. Finally, the paper overall offers a model of negotiating agents that integrates previous work on information-based agency and that overcomes some limitations of utility-based architectures (e.g. preference elicitation or valuing information).

References

1. J. L. Arcos, M. Esteva, P. Noriega, J. A. Rodríguez, and C. Sierra. Environment engineering for multiagent systems. *Journal on Engineering Applications of Artificial Intelligence*, 18, 2005.
2. J. Halpern. *Reasoning about Uncertainty*. MIT Press, 2003.
3. M. Jaeger. Representation independence of nonmonotonic inference relations. In *Proceedings of KR'96*, pages 461–472. Morgan Kaufmann, 1996.
4. J. J. Kelly. A new interpretation of information rate. *IEEE Transactions on Information Theory*, 2(3):185–189, September 1956.
5. P. Klemperer. What really matters in auction design. *The Journal of Economic Perspectives*, 16(1):169–189, 2002.
6. D. Lawrence. *The Economic Value of Information*. Springer-Verlag, 1999.
7. A. H. Maslow. A theory of human motivation. *Psychological Review*, 50:370–396, 1943.
8. C. Sierra and J. Debenham. Trust and honour in information-based agency. In P. Stone and G. Weiss, editors, *Proceedings Fifth International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2006*, pages 1225–1232, Hakodate, Japan, May 2006. ACM Press, New York.
9. C. Sierra and J. Debenham. Information-based agency. In *Proceedings of Twentieth International Joint Conference on Artificial Intelligence IJCAI-07*, pages 1513–1518, Hyderabad, India, January 2007.
10. C. Sierra and J. Debenham. The LOGIC Negotiation Model. In *Proceedings Sixth International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2007*, Honolulu, Hawai'i, May 2007.

Teaching Autonomous Agents to Move in a Believable Manner within Virtual Institutions

A. Bogdanovych, S. Simoff, M. Esteva, and J. Debenham

Abstract Believability of computerised agents is a growing area of research. This paper is focused on one aspect of believability - believable movements of avatars in normative 3D Virtual Worlds called Virtual Institutions. It presents a method for implicit training of autonomous agents in order to "believably" represent humans in Virtual Institutions. The proposed method does not require any explicit training efforts from human participants. The contribution is limited to the lazy learning methodology based on imitation and algorithms that enable believable movements by a trained autonomous agent within a Virtual Institution.

1 Introduction

With the increase of the range of activities and time humans spend interacting with autonomous agents in various computer-operated environments comes the demand for believability in the behaviour of such agents. These needs span from the booming game industry, where developers invest their efforts in smart and absorbingly believable game characters, to inspiring shopping assistants in the various areas of contemporary electronic commerce.

Existing research in the field of believable agents has been focused on imparting rich interactive personalities [1]. Carnegie-Mellon set of requirements for believable agents include personality, social role awareness, self-motivation, change, social re-

A. Bogdanovych, S. Simoff
University of Western Sydney, Australia, e-mail: {a.bogdanovych, s.simoff}@uws.edu.au

M. Esteva
Artificial Intelligence Research Institute (IIIA), Barcelona, Spain, e-mail: esteva@iiia.csic.es

J. Debenham
University of Technology, Sydney, Australia, e-mail: debenham@it.uts.edu.au

Please use the following format when citing this chapter:

Bogdanovych, A., Simoff, S., Esteva, M. and Debenham, J., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 5564.

lationships, and "illusion of life". The research in building mo-dels of different features that contribute to believability utilises the developments in cognitive modeling and attempts to formalise those models in computational form to implement them in virtual environments [2].

Integrating these believability characteristics into virtual environments (i) is associated with computational and architectural complexity; (ii) is platform and problem dependent, and (iii) is essentially far from achieving a high level of believability [3]. In order to address these drawbacks, rather than identifying, modeling and implementing different characteristics of believability some researchers investigate the automated approach of learning by imitation [4]. Imitation learning is most effective in environments where the actions of a human principal are fully observable and are easy to interpret by the agent [5]. Virtual Worlds where both humans and agents are fully immersed are quite efficient in terms of human observation facilities [5]. Even better means of observation are offered by Virtual Institutions [6] - a new class of normative Virtual Worlds, that combine the strengths of Virtual Worlds and normative multi-agent systems, in particular, electronic institutions [7]. In this "symbiosis" Virtual Worlds provide the visual interaction space and Electronic Institutions enable the rules of interaction. The environment assumes similar embodiment for all participants, so every action that a human performs can be observed and reproduced by an agent, without a need to overcome the embodiment dissimilarities. Moreover, the use of Electronic Institutions provides context and background knowledge for learning, helping to explain the tactical behavior and goals of the humans.

Further in the paper we outline the learning method called "implicit training". The explanation of this method and its role within Virtual Institutions is structured as follows. Section 2 outlines the basics of Virtual Institutions technology. Section 3 presents the principles of the implicit training method, with the implementation details given in Section 4. Section 5 describes the experimental results on learning to move in believable manner. Section 6 concludes the paper.

2 Virtual Institutions

Virtual Institutions are 3D Virtual Worlds with normative regulation of participants' interactions [6]. The development of such Virtual Worlds is separated into two phases: specification of the institutional rules and design of the visualization. The specification defines which actions require institutional verification while the rest of the actions are assumed to be safe and can be instantly performed. Rule specification utilises the "Electronic Institutions" methodology [7], which provides facilities for formalizing the interactions of participants through interaction protocols and runtime infrastructure that ensures the validity of the specified rules and their correct execution. The rules of a Virtual Institution are determined by three types of conventions (for detailed explanation see [7]):

1. *Conventions on language* form the *Dialogical Framework* dimension. It determines language ontology and illocutionary particles that agents should use, roles they can play and the relationships or incompatibilities among the roles.

2. *Conventions on activities* form the *Performative Structure* dimension. It determines in which types of dialogues agents can engage during the activities they perform in a scene, which protocols to use in the dialogues, which sublanguage of the overall institutional language can be used in each scene, and which conventions regulate the in and out flux of agents in scenes. Scenes are interconnected through "transitions" to form a network that represents sequences of activities, concurrency of activities or dependencies among them.

3. *Conventions on behavior* form the *Norms* dimension. Electronic Institutions restrict agent actions within scenes to illocutions and scene movements. Norms determine the commitments that agents acquire while interacting within an institution. These commitments restrict future activities of the agent. They may limit the possible scenes to which agents can go, and the illocutions that can henceforth be uttered.

Virtual Institutions are visualized as 3D Virtual Worlds, where a single Virtual Institution is represented as a building located inside the space labeled as "garden." The visualization is aligned with the formalised institution rules. The participants are visualized as avatars. Only participants with specified roles can enter the institutional buildings, where they can act according to the rules specification of respective institution. Each institutional building is divided into a set of rooms (every room represents a scene), which are separated by corridors (transitions) and doors. The doors are open or closed for a participant depending on the acceptance of participant's role by the corresponding scene and the execution state of the institution. Inside each of the rooms only actions that comply with the protocol of the corresponding scene can be executed (for more details see [6]).

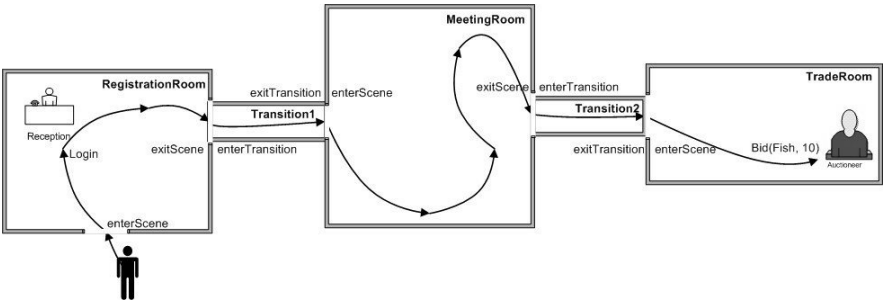


Fig. 1 Outline of a prototypical Virtual Institution containing 3 scenes.

Fig. 1 outlines a prototypical Virtual Institution containing 3 scenes - RegistrationRoom, MeetingRoom and TradeRoom, visualized as rooms connected via corridors. The actions controlled by the institution (institutional level actions) include: enterScene, exitScene, enterTransition, exitTransition and login. The rest of the actions (visual level actions) require no institutional control, these are: moving, jumping, colliding with objects, turning etc. The directed line represents the trajectory of the participant's movement. The solid figure is the participant, the rest correspond to internal agents (employees of the institution), in this case, a Receptionist and an Auctioneer. The Receptionist verifies the login and password of the participant

in the Registration Room, and unlocks the doors to other rooms if the identity of the participant is proven. The Auctioneer sells different goods in the TradeRoom. It announces the product to be auctioned, waits for incoming bids and sells it to the winner. The Meeting Room is used for social interaction between buyers. In the scenario shown in Fig. 1 the goal of the human is to buy fish in the TradeRoom.

3 Principles of Implicit Training in Virtual Institutions

Existing 3D Virtual Worlds are mostly human centered with very low agent involvement. Virtual institutions, in contrast, is an agent-centered technology, which treats humans as heterogenous, self-interested agents with unknown internal architecture. Every human participant (principal) is always supplied with a corresponding software agent, that communicates with the institutional infrastructure on human's behalf. The *couple agent/principal* is represented by an avatar. Each avatar is manipulated by either a human or an autonomous agent through an interface that translates all activities into terms of the institution machine understandable language. The autonomous agent is always active, and when the human is driving the avatar the agent observes the avatar actions and learns how to make the decisions on human's behalf. At any time a human may decide to let the agent control the avatar via ordering it to achieve some task. If the agent is trained to do so it will find the right sequence of actions and complete the task in a similar way a human would.

The training of autonomous agents in Virtual Institutions happens on both visual and institutional levels. The actions of the visual are important for capturing human-like movement. The actions of the institutional level, on the one hand, help the autonomous agent to understand when to start and stop recording the actions of the visual level and which context to assign to the recorded sequences. On the other hand, analyzing the sequence of institutional level actions helps, in a long run, to understand how to reach different rooms and separate the sequences of actions there into meaningful logical states of the agent.

Every dimension of the institutional specification contributes to the quality of learning in the following way.

Dialogical Framework: the roles of the agents enable the separation of the actions of the human into different logical patterns. The message types specified in the ontology help to create a connection between the objects present in the Virtual Worlds, their behaviors and the actions executed by the avatars.

Performative Structure: Enables grouping of human behavior patterns into actions relevant for each room.

Scene Protocols: Enable the creation of logical landmarks within human action patterns in every room.

4 Implementation of the Implicit Training Method

The implicit training has been implemented as a lazy learning method, based on graph representation. The Virtual Institution corresponds to the scenario outlined in

Fig. 1. It is visualised as a garden and an institutional building inside the garden. The institutional building consists of 3 rooms connected by corridors. Starting as an avatar in the garden, each participant can enter inside the building and continue moving through the rooms there. In our case, the participants in the institution play two different roles: receptionist and guest. The implicit training method is demonstrated on learning movement styles.

4.1 Constructing the learning graph

When a human operator enters the institution, the corresponding autonomous agent begins recording operator's actions, storing them inside a learning graph similar to the one outlined in Fig. 2. The nodes of this graph correspond to the institutional messages, executed in response to the actions of the human. Each of the nodes is associated with two variables: the message name together with parameters and the probability $P(Node)$ of executing the message. The probability is continuously updated, and in the current implementation it is calculated as follows:

$$P(Node) = \frac{n_a}{n_o} \tag{1}$$

Here n_o is the number of times a user had a chance to execute this particular message and n_a is the number of times when s/he actually did execute it.

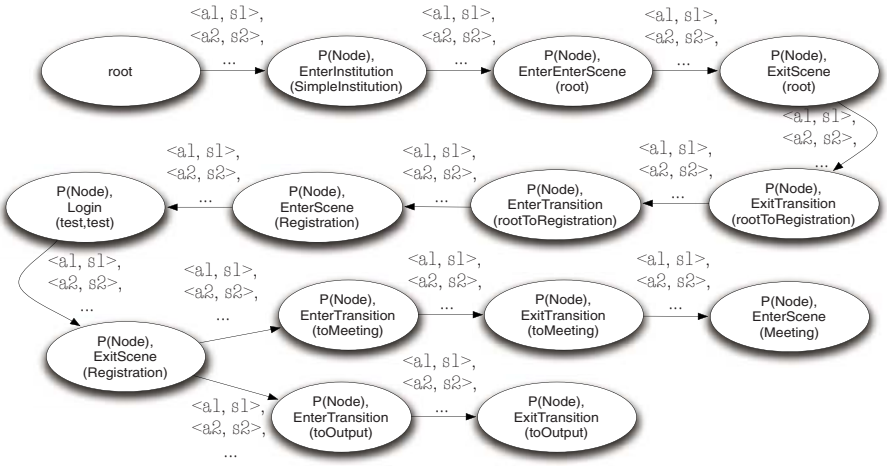


Fig. 2 A fragment of the learning graph.

The arcs connecting the nodes are associated with the prerecorded sequences of the visual level actions $(s_1, \ldots, s_n)$ and the attribute vectors that influenced them $(a_1, \ldots, a_n)$. Each pair $\langle a_n, s_n \rangle$ is stored in a hashtable, where a_i is the key of the table and s_i is the value. Each a_i consists of the list of parameters:

$$a_i = \langle p_1, \ldots \ p_k \rangle \tag{2}$$

A simplifying assumption behind the training is that the behaviour of the principle is only influenced by what is currently visible through the field of view of the avatar. We limit the visible items to the objects located in the environments and other avatars. So, the parameters used for learning are recorded in the following form:

$$p_i = \langle V_o, V_{av} \rangle \tag{3}$$

where V_o is the list of currently visible objects; V_{av} is the list of currently visible avatars. The list of the visible objects is represented by the following set:

$$V_o = \{\langle O_1, D_1 \rangle, \ldots, \langle O_j, D_j \rangle, \ldots, \langle O_m, D_m \rangle\} \tag{4}$$

where O_j are the objects that the agent is able to see from it's current position in the 3D Virtual World; D_j are the distances from the current location of the agent to the centers of mass of these objects.

The list of visible avatars is specified as follows:

$$V_{av} = \{\langle N_1, R_1, DAv_1 \rangle, \ldots, \langle N_p, R_k, DAv_p \rangle\} \tag{5}$$

Here, N_k correspond to the names of the avatars that are visible to the user, R_k are the roles played by those avatars, and DAv_k are the distances to those avatars. Each of the sequences (s_i) consists of the finite set of visual level actions:

$$s_i = \langle SA_1, SA_2, \ldots, SA_q \rangle \tag{6}$$

Each of those actions defines a discrete state of the trajectory of avatar's movement. They are represented as the following vector:

$$SA_l = \langle pos, r, h, b \rangle \tag{7}$$

where *pos* is the position of the agent, r is the rotation matrix, h is the head pitch matrix, b is the body yaw matrix. Those matrices provide the most typical way to represent a movement of a character in a 3D Virtual World.

Each time an institutional message is executed, the autonomous agent records the parameters it is currently able to observe, creates a new visual level sequence and every 50 ms adds a new visual level message into it. The recording is stopped once a new institutional message is executed.

4.2 Applying the learning graph

Once the learning graph is completed an agent can accept commands from the principal. Each command includes a special keyword "Do:" and a valid institutional level message, e.g. "Do:EnterScene(Meeting)". The nodes of the learning graph are seen as internal states of the agent, the arcs determine the mechanism of switching between states and $P(Node)$ determines the probability of changing the agent's cur-

rent state to the state determined by the next node. Once the agent reaches a state $S(Node_i)$ it considers all the nodes connected to $Node_i$ that lead to the goal node and conducts a probability driven selection of the next node ($Node_k$). If $Node_k$ is found: the agent changes its current state to $S(Node_k)$ by executing the best matching sequence of the visual level actions recorded on the arc that connects $Node_i$ and $Node_k$. If there are no visual level actions recorded on the arc - the agent sends the message associated to $Node_k$ and updates it's internal state accordingly.

For example, let the agent need to reach the state in the learning graph expressed as "$S(EnterInstitution(SimpleInstitution))$". To achieve this it has to select and execute one of the visual level action sequences stored on the arc between the current node and the desired node of the learning graph. The parameters of this sequence must match the current situation as close as possible. To do so the agent creates the list of parameters it can currently observe and passes this list to a classifier (currently, a nearest neighbor classifier [8]). The later returns the best matching sequence and the agent executes each of its actions. The same procedure continues until the desired node is reached.

5 Experiments on Learning Believable Movement

During 10 sessions we have trained an autonomous agent to believably act in the institution from Fig. 1. We started recording the actions of the human playing the "guest" role in the garden, facing the avatar towards different objects and having the receptionist agent located in various positions. In each training session the trajectory was easily distinguishable given the observed parameters.

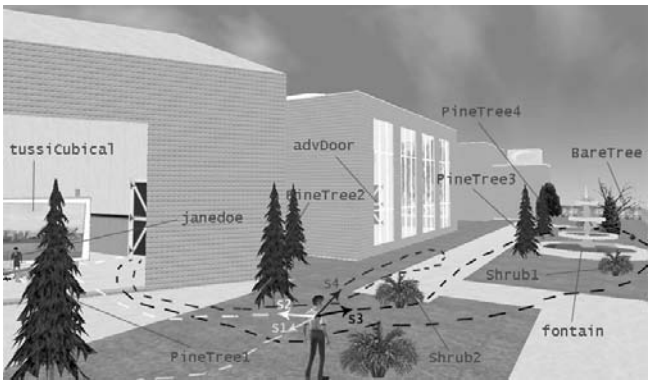


Fig. 3 Training the agent in the garden.

Fig. 3 gives an impression of how the training was conducted. It shows fragments of the 4 different trajectories (displayed as dotted lines - $S_1 ... S_4$) generated by the "guest" avatar. The arrows marked with $S_1 ... S_4$ correspond to the direction of view of the avatar at the moment when the recording was initiated. The location of the receptionist agent, it's role and position together with the objects located in the en-

vironment and distances to them were the parameters used for learning. In Fig. 3 the objects include “Pinetree”, “tussiCubical”, “advDoor” and the receptionist, showed in the figure as “janedoe”.

Table 1 Parameters used in the training session.

parameterID	parameterName	possibleValues
p_1	janedoeRole	{recep,null,guest}
p_2	DISTadvDoor	numeric
p_3	DISTPineTree4	numeric
p_4	DISTBareTree	numeric
p_5	SEEfontain	{y,n}
p_6	DISTtussiCubical	numeric
p_7	DISTPineTree3	numeric
p_8	DISTPineTree2	numeric
p_9	DISTPineTree1	numeric
p_{10}	SEEtussiCubical	{y,n}
p_{11}	SEEBareTree	{y,n}
p_{12}	SEEadvDoor	{y,n}
p_{13}	SEEshrub3	{y,n}
p_{14}	SEEshrub2	{y,n}
p_{15}	DISTfontain	numeric
p_{16}	SEEshrub1	{y,n}
p_{17}	DISTshrub3	numeric
p_{18}	SEEPineTree4	{y,n}
p_{19}	DISTshrub2	numeric
p_{20}	SEEPineTree3	{y,n}
p_{21}	DISTshrub1	numeric
p_{22}	SEEPineTree2	{y,n}
p_{23}	SEEPineTree1	{y,n}
p_{24}	SEEjanedoe	{y,n}
p_{25}	DISTjanedoe	numeric

The agent has been trained to enter the Meeting Room. The resultant learning graph was similar to the one in Fig. 2. Table 1 presents the list of all parameters stored in the graph on the arc between “root” and “EnterInstitution(SimpleInstitution)” nodes. Parameters, having names beginning with: (i) “SEE” correspond to the objects or avatars that were appearing in the field of view of the user at the moment of recording; (ii) “DIST” correspond to the distance measure between the user and the center of mass of a visible object. The distance to objects not visible is equal to zero. Parameter “janedoeRole” defines the role of the receptionist agent “janedoe”. When the receptionist was not visible in the field of view of the trained guest agent, the values of janedoeRole is “null”. When the “jandoe” was located outside the institution it’s role was “guest” and inside the registration room it was “receptionist”.

Table 2 A fragment of the data used in the training session.

Nr	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	p_{16}	p_{17}	p_{18}	p_{19}	p_{20}	p_{21}	p_{22}	p_{23}	p_{24}	p_{25}	*S*
1	null	0	0	0	n	0	0	0	31	n	n	n	n	n	0	n	0	n	0	n	0	n	y	n	0	S_1
2	recep	0	0	0	n	55	0	0	39	y	n	n	n	n	0	n	0	n	0	n	0	n	y	y	67	S_2
3	null	0	0	0	y	0	61	0	0	n	n	n	n	n	96	y	0	n	0	y	70	n	n	n	0	S_3
4	null	74	0	0	y	0	61	0	0	n	n	y	n	n	95	y	0	n	0	y	67	n	n	n	0	S_4
5	recep	0	0	0	n	56	0	0	0	y	n	n	n	n	0	n	0	n	0	n	0	n	n	y	68	S_5
6	null	0	0	0	n	0	0	43	77	n	n	n	n	y	0	n	0	n	48	n	0	y	y	n	0	S_6
7	guest	0	0	0	y	0	0	0	0	n	n	n	n	n	96	y	0	n	0	n	69	n	n	y	24	S_7
8	null	0	0	24	n	0	0	0	0	n	y	n	y	n	0	n	42	n	0	n	0	n	n	n	0	S_8
9	null	0	0	0	n	0	0	0	0	n	n	n	n	n	0	n	0	n	0	n	0	n	n	n	0	S_9
10	guest	0	41	0	y	0	96	0	0	n	n	n	n	n	69	n	0	y	0	y	0	n	n	y	38	S_{10}

Table 2 presents the training data stored on the arc between “root” and “EnterInstitution(SimpleInstitution)” during 10 recording sessions along the parameters, listed in Table 1. The “S” column shows the acronyms of the sequences of actions of the visual level of execution. The first four sequences correspond to the trajectories $S_1 ... S_4$ outlined in Fig. 3.

Each of the tests was conducted as follows. Two operators entered the Virtual World by two different avatars: “janedoe” (driven by one of the researchers in our lab), playing the “receptionist” or “guest” role and “agent0” (controlled by an independent observer), always playing the “guest” role. Both avatars were positioned in various locations and the avatar “agent0” was facing a selected direction (with janedoe either visible or not). On the next step agent0 was instructed to leave the garden, enter the institution, walk into the registration room, exit it and then walk through the next transition to the Meeting Room. The agent then looked for the right sequence of the institutional level actions, which in the given case were: EnterInstitution(SimpleInstitution), EnterScene(root), ExitScene(root), EnterTransition(rootToRegistration), ExitTransition(rootToRegistration), EnterScene(Registration), Login(test, test), ExitScene(Registration), EnterTransition(toMeeting), ExitTransition(toMeeting), EnterScene(Meeting). To execute those actions the agent needed to launch the appropriate sequence of the visual level actions, stored on the arcs of the learning graph. The classifier was given the list of currently observed parameters as the input and as the output it returned the sequence that was supposed to fit best. After completion of recording, we conducted a series of 20 tests to check whether the trained agent would act in a believable manner. Table 3 presents the experiments’ results.

Table 3 Classifier performance: input data and recommendations.

Nr	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}	p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	p_{16}	p_{17}	p_{18}	p_{19}	p_{20}	p_{21}	p_{22}	p_{23}	p_{24}	p_{25}	S
11	guest	0	0	0	n	0	0	0	0	n	n	n	n	y	0	y	0	n	40	n	91	n	n	y	36	S_7
12	null	0	0	0	n	59	0	0	36	y	n	n	n	n	0	n	0	n	0	n	0	n	y	n	0	S_1
13	guest	0	36	0	y	0	91	0	0	n	n	n	n	n	75	y	0	y	0	y	63	n	n	y	17	S_{10}
14	null	0	0	0	y	0	0	0	0	n	n	n	n	n	36	y	0	n	0	n	82	n	n	n	0	S_3
15	null	0	0	77	y	0	0	0	0	n	y	n	y	n	30	n	94	n	0	n	0	n	n	n	0	S_8
16	guest	0	0	0	n	0	0	0	0	n	n	n	n	n	0	n	0	n	0	n	0	n	n	y	25	S_9
17	guest	0	0	86	n	0	0	0	0	n	y	n	y	n	0	n	77	n	0	n	0	n	n	y	34	S_8
18	null	65	0	0	y	0	51	0	0	n	n	y	n	n	87	n	0	n	0	y	0	n	n	n	0	S_4
19	recep	0	0	0	n	41	0	0	0	y	n	n	n	n	0	n	0	n	0	n	0	n	n	y	51	S_5
20	recep	0	0	0	n	72	0	41	40	y	n	n	n	n	0	n	0	n	0	n	0	y	y	y	78	S_2
21	guest	0	0	0	n	0	0	63	95	n	n	n	n	y	0	n	0	n	42	n	0	y	y	y	18	S_6
22	null	0	0	0	y	0	18	0	0	n	n	n	n	n	50	y	0	n	0	y	99	n	n	n	0	S_3
23	null	0	0	0	n	0	0	0	0	n	n	n	n	n	0	n	0	n	0	n	0	n	n	n	0	S_9
24	guest	0	95	0	y	0	0	0	0	n	n	n	n	n	71	y	0	y	0	n	41	n	n	y	14	S_7
25	guest	0	0	0	y	0	26	0	0	n	n	n	n	y	59	y	0	n	82	y	84	n	n	y	71	S_7
26	null	0	0	0	y	0	26	0	0	n	n	n	n	y	59	y	0	n	82	y	84	n	n	n	0	S_3
27	null	0	0	43	n	0	0	0	0	n	y	n	y	n	0	n	60	n	0	n	0	n	n	n	0	S_8
28	null	0	45	0	y	0	96	0	0	n	n	n	n	n	58	n	0	y	0	y	0	n	n	n	0	S_{10}
29	null	0	0	0	n	0	0	0	0	n	n	n	y	n	0	n	63	n	0	n	0	n	n	n	0	S_9
30	guest	0	0	0	n	0	0	0	0	n	n	n	y	n	0	n	63	n	0	n	0	n	n	y	14	S_9

Fig. 4 shows the eye direction of the guest and the positions of both avatars. Solid dots marked with the number of experiment in the figure correspond to the positions of the guest. The arrows represent guest’s eye direction. The female figure marked with the experiment number shows the positions of the receptionist (when it was visible to the guest). The experiment numbers in Fig. 4 correspond to the ones specified in the “Nr” columns in Table 2 and Table 3. The numbers 1-10 are the initial recordings and 11-30 represent the conducted experiments. The “S” column in Table 3 outlines the acronyms of the action sequences (as used in Table 2) executed by the agent as a result of classifier’s recommendation.

In every test the believability of the movement was assessed by an independent observer. In all cases it was evaluated as believable.

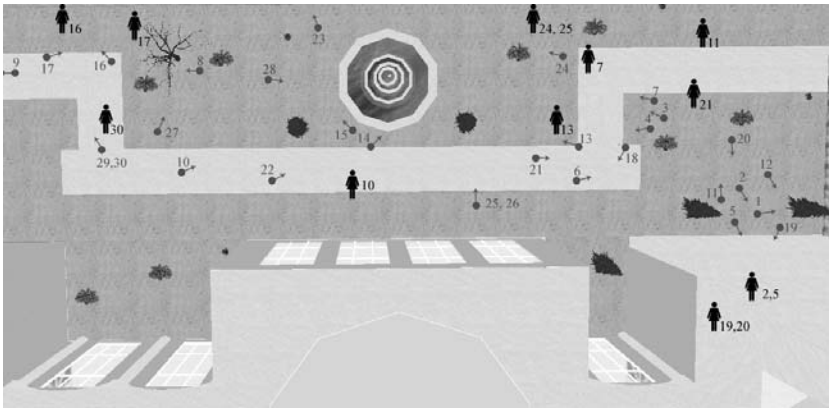


Fig. 4 Design of the experiments in the institution space.

6 Conclusion

We have presented the concept of implicit training used for teaching human behavioral characteristics to autonomous agents in Virtual Institutions. The developed prototype and conducted experiments confirmed the workability of the selected learning method and the validity of the implicit training concept.

Acknowledgements. This research is partially supported by an ARC Discovery Grant DP0879789 and the e-Markets Research Program (http://e-markets.org.au).

References

1. Loyall, A.B.: Believable agents: building interactive personalities. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA (1997)
2. Magnenat-Thalmann, N., Kim, H., Egges, A., Garchery, S.: Believability and Interaction in Virtual Worlds. In: Proceedings of the Multimedia Modeling Conference (MMM'05), Washington, DC, USA, IEEE Computer Society (2005) 2–9
3. Livingstone, D.: Turing's test and believable AI in games. Computers in Entertainment **4**(1) (2006) 6–18
4. Breazeal, C.: Imitation as social exchange between humans and robots. In: Proceedings of the AISB Symposium. (1999) 96–104
5. Gorman, B., Thurau, C., Bauckhage, C., Humphrys, M.: Believability Testing and Bayesian Imitation in Interactive Computer Games. In: Proceedings of SAB'06 conference. Volume LNAI 4095., Springer (2006) 655–666
6. Bogdanovych, A.: Virtual Institutions. PhD thesis, University of Technology, Sydney, Australia (2007)
7. Esteva, M.: Electronic Institutions: From Specification to Development. PhD thesis, Institut d'Investigació en Intelligència Artificial (IIIA), Spain (2003)
8. Hastie, T., Tibshirani, R.: Discriminant adaptive nearest neighbor classification and regression. In Touretzky, D.S., Mozer, M.C., Hasselmo, M.E., eds.: Advances in Neural Information Processing Systems. Volume 8., The MIT Press (1996) 409–415

DATA MINING

Mining Fuzzy Association Rules from Composite Items

M. Sulaiman Khan[1], Maybin Muyeba[2], and Frans Coenen[3]

Abstract This paper presents an approach for mining fuzzy Association Rules (ARs) relating the properties of composite items, i.e. items that each feature a number of values derived from a common schema. We partition the values associated to properties into fuzzy sets in order to apply fuzzy Association Rule Mining (ARM). This paper describes the process of deriving the fuzzy sets from the properties associated to composite items and a unique Composite Fuzzy Association Rule Mining (CFARM) algorithm founded on the certainty factor interestingness measure to extract fuzzy association rules. The paper demonstrates the potential of composite fuzzy property ARs, and that a more succinct set of property ARs can be produced using the proposed approach than that generated using a non-fuzzy method.

1. Introduction

Association Rule Mining (ARM) is an important and well established data mining topic. The objective of ARM is to identify patterns expressed in the form of Association Rules (ARs) in transaction data sets [1]. The attributes in ARM data sets are usually binary valued but it has been applied to quantitative and categorical (non-binary) data [2]. With the latter, values can be split into ranges such that each range represents a binary valued attribute and ranges linguistically labelled; for example "low", "medium", "high" etc. Values can be assigned to these range attributes using crisp boundaries or fuzzy boundaries. The application

[1] M. Sulaiman Khan (PhD student)

School of computing, Liverpool Hope University, L16 9JD, UK email: khanm@hope.ac.uk

[2] Dr. Maybin Muyeba

School of computing, Liverpool Hope University, L16 9JD, UK email: muyebam@hope.ac.uk

[3] Dr. Frans Coenen

Department of Computer Science, University of Liverpool, L69 3BX email: frans@csc.liv.ac.uk

Please use the following format when citing this chapter:

Khan, M.S., Muyeba, M. and Coenen, F., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 67-76.

of ARM using the latter is referred to as fuzzy ARM (FARM) [3]. The objective of fuzzy ARM is then to identify fuzzy ARs. Fuzzy ARM has been shown to produce more expressive ARs than the "crisp" methods [3, 5, 8].

We approach the problem differently in this paper by introducing "Composite Item Fuzzy ARM" (CFARM) whose main objective is the generation of fuzzy ARs associating the "properties" linked with composite attributes [4], i.e., attributes or items composed of sets of sub-attributes or sub-items that conform to a common schema. For example, given an image mining application, we might represent different areas of each image in terms of groups of pixels such that each group is represented by the normalized summation of the RGB values of the pixels in that group. In this case the set of composite attributes (I) is the set of groups, and the set of properties (P) shared by the groups is equivalent to the RGB summation values (i.e. $P = \{R, G, B\}$). Another could be the market basket analysis, where I is a set of groceries, and P is a set of nutritional properties that these groceries posses (i.e. P = {Pr, Fe, Ca, Cu,..}) standing for protein, Iron etc. Note that the actual values (properties) associated with each element of I will be constant, unlike in the case of the image mining example. We note that there are many examples depending on application area but we limit ourselves to these given here.

The term composite item has been used previously in [6, 7] and defined as a combination of several items e.g. if itemset {A, B} and {A, C} are not large then rules {B}→{A} and {C}→{A} will not be generated, but by combining B and C to make a new composite item {BC} which may be large, rules such as {BC}→{A} may be generated. In this paper we define composite items differently as indicated earlier, to be an item with properties (see Sect. 3). This definition is consistent in [4] which also defines composite attributes in this manner, i.e. an attribute that comprises two or more sub-attributes.

In this paper, the concept of "Composite item" mining of property ARs is introduced, the potential of using property ARs in many applications and a demonstration of the greater accuracy produced using the certainty factor measure. In addition, it is demonstrated that a more succinct set of property ARs (than that generated using a non-fuzzy method) can be produced using the proposed approach.

The paper is organised as follows; section 2 presents a sequence of basic concepts, section 3 presents the methodology with an example application, Section 4 presents results of the CFARM approach and section 5 concludes the paper with a summary of the contribution of the work and directions for future work.

2. Problem Definition

The problem definition consists of basic concepts to define composite items, fuzzy association rule mining concepts, the normalization process for Fuzzy

Transactions (*FT)* and interestingness measures. Interested readers can see [11] for the formal definitions and more details.

To illustrate the concepts, we apply the methodology using market basket analysis where the set of groceries have a common set of nutritional quantitative properties. Some examples are given in Table 1.

Table 1 Composite items (groceries) with their associated properties (nutrients)

Items/Nutrients	Protein	Fibre	Carbohydrate	Fat	…
Milk	3.1	0.0	4.7	0.2	…
Bread	8.0	3.3	43.7	1.5	…
Biscuit	6.8	4.8	66.3	22.8	…
…	…	…	…	…	…

To illustrate the context of the problem, Table 1 shows composite edible items, with common properties (Protein, Fibre,…). The objective is then to identify consumption patterns linking these properties and so derive fuzzy ARs.

2.1 Basic Concepts

A *Fuzzy Association Rules* [8] is an implication of the form:

$$\text{if } \langle A, X \rangle \text{ then } \langle B, Y \rangle$$

where A and B are disjoint itemsets and X and Y are fuzzy sets. In our case the itemsets are made up of property attributes and the fuzzy sets are identified by linguistic labels.

A *Raw Dataset* D consists of a set of transactions $T = \{t_1, t_2, \cdots, t_n\}$, a set of composite items $I = \{i_1, i_2, \cdots, i_{|I|}\}$ and a set of properties $P = \{p_1, p_2, \cdots, p_m\}$. The "$k^{th}$" property value for the "j^{th}" item in the "i^{th}" transaction is given by $t_i[i_j[v_k]]$. An example is given in Table 2 where each composite item is represented using the notation <label, value>.

The raw dataset D (table 2) is initially transformed into a *Property Dataset* D^p (table 3) which consists of property transactions $T^p = \{t_1^p, t_2^p, \cdots, t_n^p\}$ and a set of property attributes P (instead of a set of composite items I). The value for each property attribute $t_i^p[p_j]$ (the "j^{th}" property in the "i^{th}" property transaction) has a value obtained by aggregating the numeric values for all p_j in t_i (see Table 3). Thus:

$$\text{Prop value}(t_i^p[p_j]) = \frac{\sum_{j=1}^{|t_i|} t_i[i_j[v_k]]}{|t_i|} \quad (1)$$

Table 2 Example raw dataset *D*

TID	Record
1	{<a,{2,4,6}>, <b,{4,5,3}>}
2	{<c,{1,2,5}>, <d,{4,2,3}>}
3	{<a,{2,4,6}>, <c,{1,2,5}>, <d,{4,1,3}>}
4	{<b,{4,5,3}>, <d,{4,2,3}>}

Table 3 Property data set D^p

TID	X	Y	Z
1	3.0	4.5	4.5
2	2.5	2.0	4.0
3	2.3	2.3	4.7
4	4.0	3.5	3.0

Once a property data set D^p is defined, it is then transformed into a *Fuzzy Dataset* D'. A fuzzy dataset D' consists of fuzzy transactions $T' = \{t'_1, t'_2, ..., t'_n\}$ and a set of fuzzy property attributes P' each of which has fuzzy sets with linguistic *labels* $L = \{l_1, l_2, ..., l_{|L|}\}$ (table 4). The values for each property attribute $t_i^p[p_j]$ are *fuzzified* (mapped) into the appropriate membership degree values using a membership function $\mu(t_i^p[p_j], l_k)$ that applies the value of $t_i^p[p_j]$ to a label $l_k \in L$. The complete set of fuzzy property attributes P' is given by $P \times L$.

Composite Itemset Value (CIV) table is a table that allows us to get property values for specific items. The *CIV* table for Table 2 is given in Table 5 below.

Table 4 Properties table

Property	Linguistic values		
	Low	Medium	High
X	$V_k \le 2.3$	$2.0 \le V_k \le 2.3$	$V_k \ge 3.3$
Y	$V_k \le 3.3$	$3.0 \le V_k \le 4.3$	$V_k \ge 4.1$
Z	$V_k \le 4.0$	$3.6 \le V_k \le 5.1$	$V_k \ge 4.7$

Table 5 *CIV* table

Item	Property attributes		
	X	Y	Z
A	2	4	6
B	4	5	3
C	1	2	5
D	4	2	3

Properties Table provides a mapping of property attribute values $t_i^p[p_j]$ to membership values according to the correspondence between the given values to the given linguistic labels. An example is given in Table 5 for the raw data set given in Table 2.

A property attribute set A, where $A \subseteq P \times L$, is a *Fuzzy Frequent Attribute Set* if its fuzzy support value is greater than or equal to a user supplied minimum

support threshold. The significance of fuzzy frequent attribute sets is that fuzzy association rules are generated from the set of discovered frequent attribute sets.

Fuzzy Normalisation is the process of finding the contribution to the fuzzy support value, m', for individual property attributes ($t_i^p[p_j[l_k]]$) such that a partition of unity is guaranteed. This is given by equation 2 where μ is the membership function:

$$t_i'[p_j[l_k]] = \frac{\mu(t_i^p[p_j[l_k]])}{\sum_{x=1}^{|L|} \mu(t_i^p[p_j[l_x]])} \tag{2}$$

If normalisation is not done, the sum of the support contributions of individual fuzzy sets associated with an attribute in a single transaction may no longer be unity which is undesirable.

Frequent fuzzy attribute sets are identified by calculating *Fuzzy Support* values. Fuzzy Support $(Supp_{Fuzzy})$ is calculated as follows:

$$Supp_{Fuzzy}(A) = \frac{\sum_{i=1}^{i=n} \prod_{\forall[i[l]]\in A} t_i'[i[l]]}{n} \tag{3}$$

where $A = \{a_1, a_2, ..., a_{|A|}\}$ is a set of property attribute-fuzzy set (label) pairs. A record t_i' "satisfies" A if $A \subseteq t_i'$. The individual vote per record, t_i is obtaining by multiplying the membership degree associated with each attribute-fuzzy set pair $[i[l]] \in A$.

2.2 Interestingness Measures

Frequent attribute sets with fuzzy support above the specified threshold are used to generate all possible rules. *Fuzzy Confidence* $(Conf_{Fuzzy})$ is calculated in the same manner that confidence is calculated in classical ARM:

$$Conf_{Fuzzy}(A \rightarrow B) = \frac{Supp_{Fuzzy}(A \cup B)}{Supp_{Fuzzy}(A)} \tag{4}$$

The Fuzzy Confidence measure $(Conf_{Fuzzy})$ described does not use $Supp_{Fuzzy}(B)$ but the *Certainty* measure $(Cert)$ addresses this. The certainty measure is a statistical measure founded on the concepts of *covariance (Cov)* and *variance (Var)* and is calculated as follows:

$$Cert(A \rightarrow B) = \frac{Cov(A,B)}{\sqrt{Var(A) \times Vat(B)}} \tag{5}$$

The value of certainty ranges from -1 to +1. We are only interested in rules that have a certainty value greater than 0. As the certainty value increases from 0 to 1, the more related the attributes are and consequently the more interesting the rule.

3. Methodology

To evaluate the approach, a market basket analysis data set with 600 composite edible items is used and the objective is to determine consumers' consumption patterns for different nutrients using RDA. The properties for each item comprised the 27 nutrients contained in the government sponsored RDA table (a partial list consists of Biotin, Calcium, Carbohydrate, ..., Vitamin K, Zinc). These RDA values represent a CIV table used in the evaluation. The property data set will therefore comprise $600 \times 27 = 16200$ attributes. The linguistic label set L was defined as follows L – {Very Low (VL), Low (L), Ideal (I), High (H), Very High (VH)}. Thus the set of fuzzy attributes $A = P \times L$ has $27 \times 5 = 135$ attributes. A fragment of this data (properties table) is given in Table 6.

Table 6 Fragment of market basket properties table[4]

Nutrients/	Very Low				Low				Ideal				High				Very High		
Fuzzy Ranges	Min	Core		Max	Min	Core		Max	Min	Core		Max	Min	Core		Max	Min	Core	
Fiber	0	1	10	15	10	15	20	25	20	25	30	35	30	33	38	39	35	40	…
Iron	0	6	8	12	8	12	16	18	16	18	19	20	19	20	22	23	22	23	…
Protein	0	1	15	30	10	20	35	40	35	40	60	65	60	65	75	80	75	70	…
Vitamin	0	15	150	200	150	200	300	400	300	350	440	500	440	490	550	600	550	600	…
Zinc	0	0.8	8	10	8	10	15	20	15	20	30	40	30	40	46	50	46	50	…
	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

A representative fragment of a raw data set (T), comprising edible items, is given in Table 7(a). This raw data is then cast into a properties data set (T^P) using the given CIV/RDA table to give the properties data set in Table 7(b). It is feasible to have alternative solutions here but we choose to code fuzzy sets {very

[4] Values could be in grams, milligrams, micrograms, International unit or any unit. Here Min is the minimum value i.e. α, Core is the core region β, δ and Max is the maximum value γ in the trapezoidal fuzzy membership function.

low, low, ideal, high, very high} with numbers {1, 2, 3, 4, 5} for the first nutrient (Pr), {6, 7, 8, 9, 10} for the second nutrient (Fe) etc [9]. Thus, data in Table 7(c) can be used by any binary ARM algorithm.

Table 7 (a) [a]

TID	Items
1	X, Z
2	Z
3	X,Y, Z
4	…

[a] Raw data (T)

Table 7 (b) [b]

TID	Pr	Fe	Ca	Cu
1	45	150	86	28
2	9	0	47	1.5
3	54	150	133	29.5
4	…	…	…	…

[b] Property data set (T^P)

Table 7 (c) [c]

TID	Pr	Fe	Ca	Cu
1	3	8	13	16
2	1	6	12	16
3	3	8	15	16
4	…	…	…	...

[c] Conventional ARM data set

This approach only gives us, the total support of various fuzzy sets per nutrient and not the degree of (fuzzy) support. This directly affects the number and quality of rules as stated in Sect. 4. To resolve the problem, the fuzzy approach here converts RDA property data set, Table 7(b), to linguistic values (Table 8) for each nutrient and their corresponding degrees of membership reflected in each transaction.

Table 8 Linguistic transaction file

TID	Protein (Pr)					Iron (Fe)					
	VL	L	Ideal	H	VH	VL	L	Ideal	H	VH	…
1	0.0	0.7	0.3	0.0	0.0	0.0	0.0	0.8	0.2	0.0	…
2	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	…
3	0.0	0.0	0.9	0.1	0.0	0.0	0.0	0.8	0.2	0.0	…
4	…	…	…	…	…	…	…	…	…	…	…

Table 8 shows only two nutrients, Pr and Fe (i.e. a total of 10 fuzzy sets).

4. Experimental Results

To demonstrate the effectiveness of the approach, we performed several experiments using a real retail data set [10]. The data is a transactional database containing 88,163 records and 16,470 unique items. For the purpose of the experiments we mapped the 600 item numbers onto 600 products in a real RDA table. Results in [11] were produced using synthetic dataset. In this paper, an improvement from [11] is that we have used real dataset in order to demonstrate the real performance of the proposed approach and algorithm.

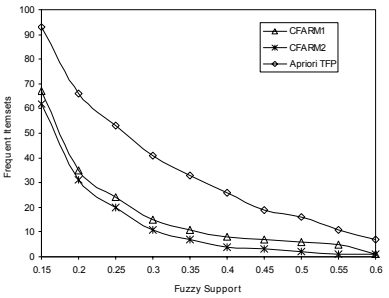
The Composite Fuzzy ARM (CFARM) algorithm is a *breadth first traversal* ARM algorithm, uses tree data structures and is similar to the Apriori algorithm

[1]. The CFARM algorithm consists of several steps. For more details on algorithm and pseudo code please see [11].

4.1 Quality Measures

In this section, we compare Composite Fuzzy Association Rule Mining (CFARM) approach against standard Quantitative ARM (discrete method) with and without normalisation. We compare the number of frequent sets and the number of rules generated using both the confidence and the certainty interestingness measure. Fig. 1 demonstrates the difference between the numbers of frequent itemsets generated using Quantitative ARM approach with discrete intervals and CFARM with fuzzy partitions. CFARM1 uses data without normalisation and CFARM2 uses normalised data. For standard Quantitative ARM, we used Apriori-TFP algorithm [12]. As expected the number of frequent itemsets increases as the minimum support decreases.

Figure 1 Number of frequent Itemsets generated using fuzzy support measures



It is clear from the results that the algorithm that uses discrete intervals produces more frequent itemsets than fuzzy partitioning method. This is because standard ARM (using discrete intervals) generates numerous artificial patterns resulting from the use of crisp boundaries. Conversley, fuzzy partitioning methods generate more accurately the true patterns in the data set due to the fact that it considers actual contribution of attributes in different intervals. CFARM2 produces comparatively less frequent itemsets than CFARM1, because the average contribution to support counts per transaction is greater without using normalization than with normalization.

Fig. 2 shows the number of rules generated using user specified fuzzy confidence. Fig. 3 shows the number of interesting rules generated using certainty measures values. Certainty measures (Fig. 2) generate fewer, but arguably better, rules than the confidence measure (Fig. 2). In both cases, CFARM2 generates less rules as compared to CFARM1; this is a direct consequence of the fact that CFARM 2 generates fewer frequent itemsets due to using normalised data.

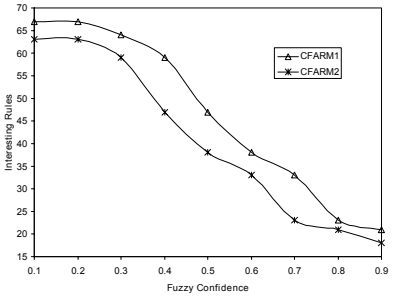


Figure 2 Interesting Rules using confidence

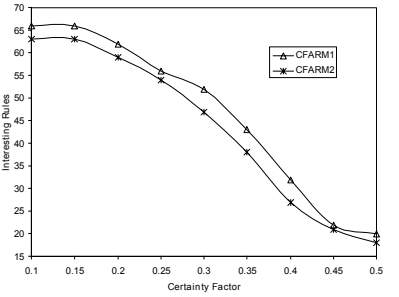


Figure 3 Interesting Rules using certainty

In addition, the novelty of the approach is its ability to analyse datasets comprised of composite items where each item has a number of property values such as the nutritional property values used in the application described here.

4.2 Performance Measures

For performance measures, we investigated the effect on algorithm execution time caused by varying the number of attributes and the data size with and without normalization using a support 0.3, confidence 0.5 and certainty 0.25. The dataset was partitioned into 9 equal partitions labelled 10K, 20K, …, 90K to obtain different data sizes We used all 27 nutrients.

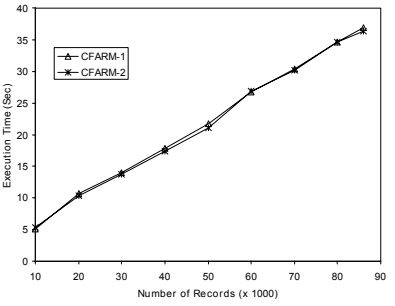


Figure 4 Execution time: No. of Records

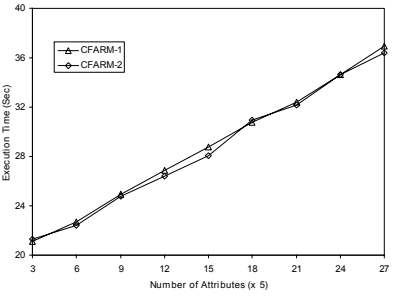


Figure 5 Execution time: No. of Attributes

Fig. 4 shows the effect on execution time by increasing the number of records. From Fig. 4 it can be seen that both algorithms have similar timings while the execution time increasing with the number of records. Fig. 5 shows the effect on execution time by varying numbers of attributes. Each property attribute has 5 fuzzy sets associated to it, therefore using 27 attributes, we have 135 columns.

However the experiments also show that the CFARM algorithm scales linearly with the number of records and attributes.

5. Conclusion and Future Work

A novel approach was presented for extracting fuzzy association rules from so-called composite items where such items have properties defined as quantitative (sub) itemsets. The properties are then transformed into fuzzy sets. The CFARM algorithm produces a more succinct set of fuzzy association rules using fuzzy measures and certainty as the interestingness measure and thus presents a new way for extracting association rules from items with properties. This is different from normal quantitative ARM. We also showed the experimental results with market basket data where edible items were used with nutritional content as properties. Of note is the significant potential to apply CFARM to other applications where items could have composite attributes even with varying fuzzy sets between attributes. We have shown that we can analyse databases with composite items using a fuzzy ARM approach.

References

1. Rakesh Agrawal, Ramakrishnan Srikant, Fast Algorithms for Mining Association Rules in Large Databases, VLDB, (1994) 487-499
2. R. Srikant and R. Agrawal, Mining Quantitative Association Rules in Large Relational Tables, Proc. ACM SIGMOD Conf. on Management of Data. ACM Press, (1996) 1-12
3. Kuok, C., Fu, A., Wong, H.: Mining Fuzzy Association Rules in Databases, ACM SIGMOD Record Vol. 27, (1), (1998) 41-46
4. W. Kim, E. Bertino and J. Garza, Composite objects revisited, ACM SIGMOD Record, Vol. 18, (2) (1989) 337-347
5. Dubois, D. E. Hüllermeier, H. Prade, A Systematic Approach to the Assessment of Fuzzy Association Rules, DM and Knowledge Discovery Journal, Vol. 13(2), (2006) 167-192
6. X. Ye and J. A. Keane, Mining Composite Items in Association Rules, Proc. IEEE Int. Conf. on Systems, Man and Cybernetics, (1997) 1367-1372
7. K. Wang, J. K. Liu and W. Ma, Mining the Most Reliable Association Rules with Composite Items, in Proc. ICDMW'06, (2006), 749-754
8. M. Delgado, N. Marin, D. Sanchez and M. A. Vila, Fuzzy Association Rules, General Model and Applications, IEEE Transactions on Fuzzy Systems, 11(2) (2003) 214-225
9. M. Sulaiman Khan, M. Muyeba, F. Coenen, On Extraction of Nutritional Patterns (NPS) Using Fuzzy Association Rule Mining, proc. Intl. Conf. on Health Informatics, Madeira, Portugal, (2008), Vol. 1, 34-42
10. Brijs T. and et al., The use of association rules for product assortment decisions: a case study, proc. 5th Intl. Conf. on Knowledge Discovery and Data Mining (1999) 254-260
11. M. Sulaiman Khan, M. Muyeba, F. Coenen, A Framework for Mining Fuzzy Association Rules from Composite Items, to appear in ALSIP (PAKDD) 2008, Osaka, Japan
12. Coenen, F., Leng, P., Goulbourne, G.: Tree Structures for Mining Association Rules, Data Mining and Knowledge Discovery, Vol. 8, No. 1, (2004) 25-51

P-Prism: A Computationally Efficient Approach to Scaling up Classification Rule Induction

Frederic T. Stahl, Max A. Bramer, and Mo Adda

Abstract Top Down Induction of Decision Trees (TDIDT) is the most commonly used method of constructing a model from a dataset in the form of classification rules to classify previously unseen data. Alternative algorithms have been developed such as the Prism algorithm. Prism constructs modular rules which produce qualitatively better rules than rules induced by TDIDT. However, along with the increasing size of databases, many existing rule learning algorithms have proved to be computational expensive on large datasets. To tackle the problem of scalability, parallel classification rule induction algorithms have been introduced. As TDIDT is the most popular classifier, even though there are strongly competitive alternative algorithms, most parallel approaches to inducing classification rules are based on TDIDT. In this paper we describe work on a distributed classifier that induces classification rules in a parallel manner based on Prism.

1 Introduction

Scaling up data mining algorithms to massive datasets has never been more topical. That is because of the fast and continuous increase in the number and size of databases. For example in the area of Molecular Dynamics (MD), simulations are conducted which describe the unfolding and folding of proteins. These simulations generate massive amounts of data which researchers are just starting to manage to store [7]. For example one single experiment can generate datasets of

Dipl.-Ing.(FH)Frederic T. Stahl
University of Portsmouth, Lion Terrace, PO1 3HE Portsmouth e-mail: Frederic.Stahl@port.ac.uk

Prof. Max A. Bramer
University of Portsmouth, Lion Terrace, PO1 3HE Portsmouth e-mail: Max.Bramer@port.ac.uk

Dr. Mo Adda
University of Portsmouth, Lion Terrace, PO1 3HE Portsmouth e-mail: Mo.Adda@port.ac.uk

Please use the following format when citing this chapter:

Stahl, F.T., Bramer, M.A. and Adda, M., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 7786.

100s of gigabytes [6]. Researchers in the MD community wish to apply data mining algorithms on MD experimental data such as pattern detection, clustering or classification [4]. However, most data mining algorithms do not scale well on such massive datasets and thus researchers are forced to sample the data to which they want to apply their data mining algorithms. Catlett's work [2] shows that sampling of data results in a loss of accuracy in the data mining result. However, Catlett conducted his experiments 16 years ago and referred to data samples that were much smaller than those nowadays. Frey and Fisher [8] showed that the rate in increase of accuracy slows down with the increase of the sample size.

However, scaling up is also an issue in applications that are concerned with the discovery of knowledge from large databases rather than predictive modelling. For instance researchers are interested in discovering knowledge from gene expression datasets, for example concerning knowledge about the influence of drugs on the gene expression levels of cancer patients. A drug might be designed to suppress tumour promoting genes, so called oncogenes. In some cases the same drug might also suppress genes that are not directly related to the tumour and thus cause adverse effects which might be lethal in rare cases. If we would sample here, we might lose data that may lead to the detection of rules that might identify a risk in applying a certain drug. Furthermore not only the number of examples but also the number of attributes which describe each example contributes to the size of the dataset [5]. For example gene expression datasets often comprise thousands or even tens of thousands of genes which represent attributes in a relational data table.

We present work on a parallel data distributed classifier based on the Prism [3] algorithm. We expect to be able to induce qualitatively good rules with a high accuracy and a sufficient scale up on massive datasets, such as gene expression datasets.

2 Inducing Modular Classification Rules Using Prism

The Prism classification rule induction algorithm promises to induce qualitatively better rules compared with the traditional TDIDT algorithm. According to Cendrowska, that is because Prism induces modular rules that have fewer redundancies compared with TDIDT [3]. Rule sets such as:

IF a = 1 AND b = 1 THEN CLASS = 1
IF c = 1 AND d = 1 THEN CLASS = 2

which have no common variable cannot be induced directly by TDIDT [3]. Using TDIDT would produce unnecessarily large and confusing decision trees. Cendrowska presents the Prism algorithm [3] as an alternative to decision trees. We implemented a version of Prism that works on continuous datasets like gene expression data [11]. The basic Prism algorithm, for continuous data only, can be summarised as shown in figure 1, assuming that there are $n(>1)$ possible classes. The aim is to generate rules with significantly fewer redundant terms than those derived from decision trees. Compared with decision trees Prism [1]:

- Is less vulnerable to clashes
- Has a bias towards leaving a test record unclassified rather than giving it a wrong classification
- Often produces many fewer terms than the TDIDT algorithm if there are missing values in the training set.

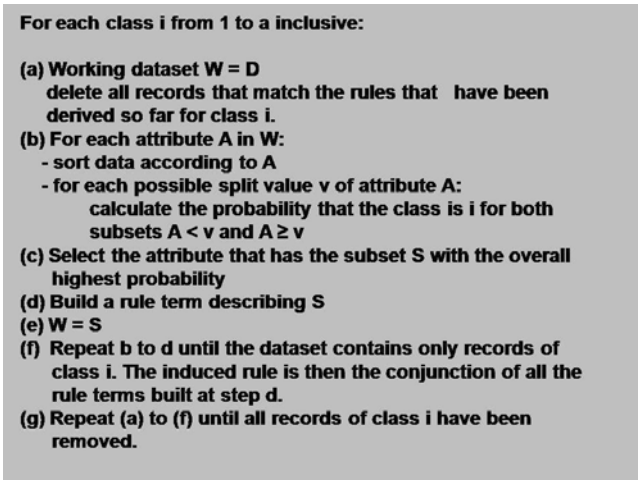


Fig. 1 The basic Prism algorithm for continuous data comprises five nested loops. The innermost loop involves sorting of the data for every continuous attribute.

However as shown in the algorithm in figure 1, the computational requirements of Prism are considerable, as the algorithm comprises five nested loops. The innermost loop involves sorting (contained in step b) the data for every continuous attribute [11]. Loosely speaking Prism produces qualitatively strong rules but suffers from its high computational complexity.

We have removed the innermost loop by pre-sorting the data once at the beginning. We did that by representing the data in the form of sorted attribute lists. Building of attribute lists is performed by decoupling the data into data structures of the form

$$\langle record\,id, attribute\,value, class\,value \rangle$$

for each attribute. Attribute lists were first introduced and successfully used in the SPRINT (Scalable PaRallelizable INduction of decision Trees) project for the parallelisation of TDIDT [10]. The use of sorted attribute lists enabled us to keep the data sorted during the whole duration of the Prism algorithm. By doing so we achieved a speedup factor of 1.8 [11].

The left-hand side of figure 2 illustrates the building of attribute lists. Note that all lists are sorted and all lists comprise a column with identifiers (id) added so that data records split over several lists can be reconstructed. As Prism removes attribute lists that are not covered by the previously induced rule term, our classifier

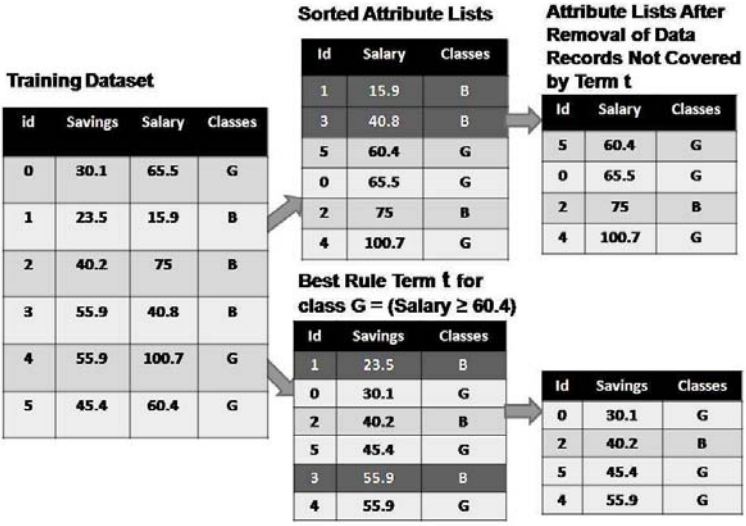


Fig. 2 The left hand side shows how sorted attribute lists are built and the right hand side shows how list records, in this case records with the ids 1 and 3, are removed in Prism.

needs to remove list records in an analogous way. For example if Prism finds a rule term $(salary \geq 60.4)$ for class G then Prism would remove the list records with the id values 1 and 3 as they are not covered by this rule. Note that the resulting list records are still sorted. This fact eliminates multiple sorting of attribute lists. The use of attribute lists in Prism enables us to parallelise the algorithm in a shared nothing environment, where each CPU has its own private memory, by data distribution [11].

3 Speeding up Prism by Parallelisation via a Distributed Blackboard System

A blackboard system is a software architecture that simulates a group of experts in front of a blackboard which have expertise in different areas. These experts communicate by reading new information from the blackboard, deriving new knowledge from it and writing this new information again on to the blackboard, thus making it accessible to the other experts. In the software architecture the blackboard is based on a server/client model. The server functions as a blackboard and the clients as experts. We are using an implementation of a distributed blackboard system developed by the Nottingham Trent University [9]. In a similar way to a shared memory version of SPRINT [13] we want to parallelise Prism by distributing 1/k chunks of each attribute list to k different expert machines. We want to synchronise the algorithm then by using the distributed blackboard system. Thus each expert machine will hold

a different part of the data and derive new knowledge from it in the form of a locally best rule term. Then each expert machine will exchange quality information about all locally best rule terms via the blackboard with the other expert machines.

4 Parallel Prism: Basic Architecture And Algorithm

As described in the previous section, the first step is to build attribute lists and distribute them to all expert machines. In the context of Parallel Prism (P-Prism), we refer to the expert machines as Worker Machines as the purpose of parallelising Prism is to split the workload, determined by the size of the training data, over k CPUs.

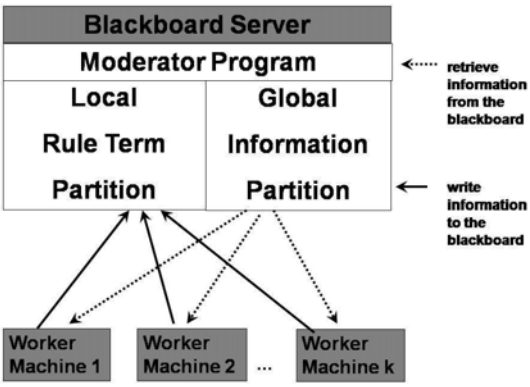


Fig. 3 Architecture of P-Prism using a blackboard server comprising two partitions, a partition for submitting rule terms to the blackboard (Local Rule Term Partition) and one to advertise global information (global information partition) to the worker machines. The moderator program on the blackboard derives the global information.

Figure 3 illustrates the basic architecture of P-Prism using a blackboard server which comprises two partitions, a partition for submitting rule terms to the blackboard, the "Local Rule Term Partition" and one to advertise Global information to the worker machines, the "Global Information Partition". Figure 4 depicts the basic P-Prism algorithm. Each worker machine M induces independently a rule term t_M for class i which is the best rule term to describe i on the local data on M. The quality of t_M is measured in the form of the probability P_M with which t_M covers class i on the local data. Each M submits t_M plus its associated P_M to the "Local Rule Term Partition" on the blackboard. The moderator program on the blackboard collects all t_Ms with their associated P_Ms and searches out the globally best rule term, which is the one with the highest P_M. The moderator program also provides global informa-

tion to the worker machines by writing it on to the "Global Information Partition". The global information comprises the globally best rule term or identifiers for the data covered by this rule term. Loosely speaking, global information informs the worker machines about the global state of the algorithm and thus how they shall proceed, e.g. deriving a further rule term or starting a new rule.

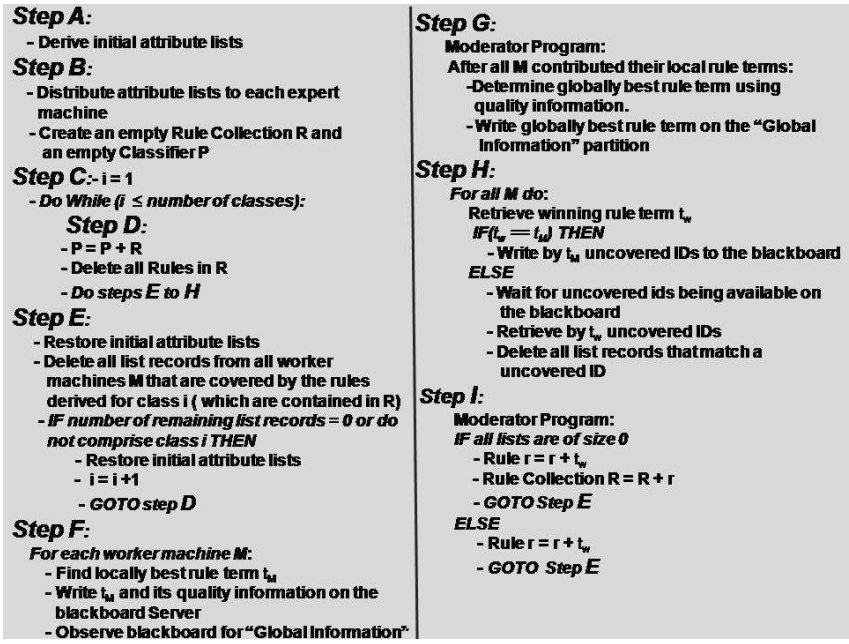


Fig. 4 Outline of the basic structure of the P-Prism algorithm. The data distribution takes place in step B and the parallelisation in step D.

Figure 4 outlines the rough structure of the proposed P-Prism algorithm. The data distribution takes place in step B by distributing the attribute lists. The parallelisation takes place in steps F to I as here every worker machine derives its local rule term and waits for the global information to become available on the blackboard.

5 Ongoing Work

So far we have set up a local area network with 4 worker machines and one blackboard server which is configured as described in section 4. The moderator program has been implemented and is fully functional. The worker machines simulate rule term induction for a complete rule in order to test the moderator program's functionality on the blackboard server. The next step is to fully implement the worker machines in order to test the computational performance of P-Prism.

5.1 Reduction of the Data Volume Using Attribute Lists

Shafer claims in his paper [10] that attribute lists could be used to buffer data to the hard disc in order to overcome memory constraints. However, buffering of attribute lists involves many I/O operations. As the data in attribute lists is even larger than the raw data, we expect a considerable slowdown of the runtime of Prism if we use buffering of attribute lists. Thus we are working on a modified version of the attribute list. As Prism mainly needs the class value distribution in a sorted attribute list, we want to reduce memory usage by building class distribution lists instead of attribute lists. Class distribution lists have the following structure:

$$\langle record\,id, class\,value \rangle$$

The class distribution list is built out of a sorted attribute list by deleting the attribute value column, thus the ids and class values in the class distribution list are sorted according to the attribute values. The total size of these lists is less than that of the raw data and even less than that of the data in the form of attribute lists. The rules induced using class distribution lists will lead to rule terms labelled with record ids instead of the actual attribute values. After all rules are induced, the record ids can easily be replaced by the actual attribute values.

The amount of memory (S) needed for Prism working on the raw data can be described by the formula $S = (8 * n + 1) * m$ bytes, where n is the number of attributes and m is the number of data records. We assume that eight bytes is the amount of memory needed to store an attribute value (assuming double precision values) and one byte to store a class value assuming a character representation. These assumptions would perfectly apply to gene expression data as a gene expression value is a double precision value. The storage needed by Prism to hold all the attribute lists in memory can be described analogously by the formula $S = (8 + 4 + 1) * n * m$ bytes. Again, the eight bytes represent an attribute value and the one byte a class value. The four byte value corresponds to an integer value for a record id in the attribute list. Representing the training data with the class distribution list structure instead of the attribute list structure eliminates the eight byte attribute values and thus only requires a memory usage of $S = (4 + 1) * n * m$ bytes [11].

However, attribute lists without the actual attribute value cannot be used exactly as stated above. We need to find a way to deal with repeated attribute values. Figure 5 illustrates the problem of repeated attribute values. The attribute list on the left hand side would find $(X \leq 2.1)$ as the rule term for the attribute X regarding class C with a covering probability of 0.67. The class distribution list on the right hand side in figure 5 represents our class distribution list with only the ids and the class values. Using only the class distribution without incorporating information about repeated attribute values for finding the best rule term would lead to a rule term of the form $(X > id0)$ for class C with a covering probability of 1. But the value of X at id 0 is 8.7, which leads to the actual rule term $(X > 8.7)$ which has a covering probability of only 0.5 as data records with ids 3 and 0 are also covered by that term. Thus we need to mark repeated attribute values in our class distribution list structure.

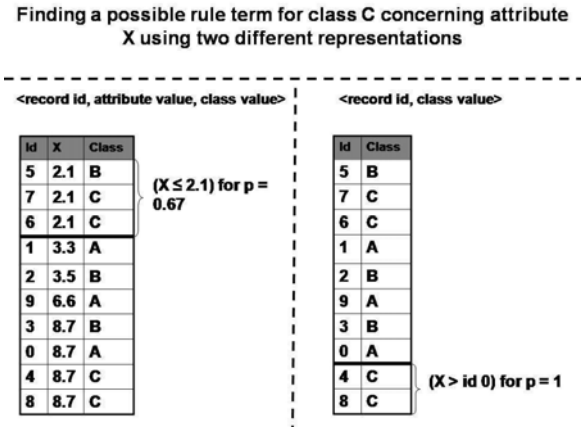


Fig. 5 Finding a possible rule term in a class distribution list without having the attribute values can lead to low wrong rule terms.

One possible way to mark repeated attribute values in the class distribution list is to add another column "indicator" which is a flag that indicates repeated values, e.g. we could use an integer that is 0 if the list record corresponds to a repeated attribute value or 1 if not. Thus the class distribution list structure would have to be altered to $\langle indicator, record\,id, class\,value \rangle$. This also leads to an altered formula for the memory usage which is $S = (4+1+1) * n * m$. The additional byte here corresponds to the added indicator. The resulting S would still be smaller than those for the raw data and the traditional attribute list structure. Concerning memory usage a better way is to use signed integers for the record id in the class distribution list structure. Positive record ids can be used for non-repeated attribute values and negative ids for repeated ones. The formula for the memory usage here would remain the same, but the disadvantage of using signed integers is that we could only represent 2^{31} data records instead of 2^{32} for unsigned integers. Another way to represent repeated attribute values without using additional memory or signed integers is using upper case and lower case characters for the class values in order to indicate repeated attribute list values. For example using lower case letters for non-repeated values and upper case letters for repeated values.

Table 1 shows the actual memory usage of Prism working with raw data, attribute lists and our class distribution list calculated using signed integers or upper case and lower case class values and using $S = (4+1) * n * m$ bytes for the memory usage of the class distribution list. The datasets are gene expression datasets concerning several diseases which can be retrieved from http://sdmc.lit.org.sg/GEDatasets/ except the SAGE-tag and Affymetix dataset. They can be found at the Gene Expression Omnibus (GEO) [12]. We clearly see that using attribute lists greatly increases the memory required to store the training data. We also see that the class distribution list outperforms the representation of data using both raw data and attribute lists in relation to memory requirements.

Table 1 Examples of the memory usage of several gene expression datasets in Prism using raw data $S1 = (8 * n + 1) * m$; using the attribute list structure $S2 = (8 + 4 + 1) * n * m$ and using our proposed class distribution list structure $S3 = (4 + 1) * n * m$. The datasets are gene expression data, thus the number of attributes is determined by the number of genes. All values for memory usage are stated in megabytes.

Dataset	Genes(n)	Examples(m)	S1	S2	S3
ALL/AML Leukaemia	7129	48	2.74	4.45	1.71
Breast cancer outcome	24481	78	15.28	24.82	9.55
CNS embryonal tumour	7129	60	3.42	5.56	2.14
Colon tumour	7129	62	3.42	5.75	2.21
Lung cancer	12533	32	3.21	5.21	2.01
Prostate cancer	12600	102	10.28	16.71	6.43
Prostate cancer outcome	12600	21	2.12	3.44	1.32
Affymetix	12332	1640	161.80	262.92	101.12
SAGE-tag	153204	243	297.83	483.97	186.14

5.2 Synchronisation

Further work will be conducted on the synchronisation of the distributed classifier. In particular several worker machines will have to wait for further information on the blackboard after they write their rule term plus its quality in the form of the covering probability on the backboard. This idle time could be used, for example, to induce locally terms for a different class value.

6 Conclusions

This paper describes work on scaling up classification rule induction on massive data sets. We first discussed why classification rule induction needs to be scaled up in order to be applicable to massive data sets and focused on a particular classification rule induction algorithm. The algorithm we focused on is the Prism algorithm. It is an alternative algorithm to decision trees which induces modular rules that are qualitatively better than rules in the form of decision trees, especially if there is noisy data or there are clashes in the dataset.

Unfortunately Prism is computationally much more expensive than decision tree induction algorithms and thus is rarely used. We described the work we did to scale up the serial version of Prism by applying presorting mechanisms to it, which resulted in a speed up factor of 1.8. We further introduced the idea of scaling up Prism by distributing the workload in the form of attribute lists over several machines in a local area network and inducing rules in parallel.

We described the basic algorithm and architecture of the parallel version of Prism which we call P-Prism. We aim to parallelise Prism by using a distributed blackboard system via which Worker Machines exchange information about their locally induced rule terms.

We further outlined how we can reduce the size of the data that needs to be held in memory by each worker machine by using class distribution lists rather than attribute lists. We described the problem that repeated attribute values will cause if we use class distribution lists, proposed 3 different ways to resolve the problem and concluded that using an upper and lower case representation of the class value is the best solution.

A further problem we briefly addressed is synchronisation, in particular the idle time of worker machines caused by waiting for global information. We propose to use this idle time to induce a rule term for a different class value in the meantime.

References

1. M. Bramer. Automatic Induction of Classification Rules from Examples Using N-Prism. In *Research and Development in Intelligent Systems XVI*. Springer, 2000.
2. J. Catlett. *Megainduction: Machine learning on very large databases*. PhD thesis, University of Technology, Sydney, 1991.
3. J. Cendrowska. PRISM: an Algorithm for Inducing Modular Rules. *International Journal of Man-Machine Studies*, 27:349–370, 1987.
4. D. Berrar, F. Stahl, C.S. Goncalves Silva, J.R. Rodrigues, R.M.M Brito, and W. Dubitzky. Towards Data Warehousing and Mining of Protein Unfolding Simulation Data. *Journal of Clinical Monitoring and Computing*, 19:307–317, 2005.
5. F. Provost and V. Kolluri. Scaling Up Inductive Algorithms: An Overview. In *Third International Conference on Knowledge Discovery and Data Mining*, pages 239–242, California, 1997.
6. F. Stahl. Systems Architecture for Distributed Data Mining on Data Warehouses of Molecular Dynamics Simulation Studies. Master's thesis, University of Applied Science Weihenstephan, 2006.
7. F. Stahl, D. Berrar, C. S. Goncalves Silva, J. R. Rodrigues, R. M. M. Brito, and W. Dubitzky. Grid Warehousing of Molecular Dynamics Protein Unfolding Data. In *Fifth IEEE/ACM Int'l Symposium on Cluster Computing and the Grid*, 2005.
8. L. J. Frey and D. H Fisher. Modelling Decision Tree Performance with the Power Law. In *eventh International Workshop on Artificial Intelligence and Statistics*, San Francisco, CA, 1999.
9. L. Nolle, K. C. P. Wong, and A. A. Hopgood. DARBS: A Distributed Blackboard System. In *Twenty-first SGES International Conference on Knowledge Based Systems*, Cambridge, 2001.
10. J. C. Shafer, R. Agrawal, and M. Mehta. SPRINT: A Scalable Parallel Classifier for Data Mining. In *Twenty-second International Conference on Very Large Data Bases*, 1996.
11. F. Stahl and M. Bramer. Towards a Computationally Efficient Approach to Modular Classification Rule Induction. In *Twenty-seventh SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, Cambridge, 2007. Springer.
12. T. Barrett. NCBI GEO: mining millions of expression profiles-database and tools. *Nucleic Acids Res.*, 33:D562–D566, 2005.
13. M. J. Zaki, C.-T. Ho, and R. Agrawal. Parallel Classification for Data Mining on Shared Memory Multiprocessors. In *Fifteenth International conference on Data Mining*, 1999.

Applying Data Mining to the Study of Joseki

Michiel Helvensteijn

Abstract Go is a strategic two player boardgame. Many studies have been done with regard to go in general, and to joseki, localized exchanges of stones that are considered fair for both players. We give an algorithm that finds and catalogues as many joseki as it can, as well as the global circumstances under which they are likely to be played, by analyzing a large number of professional go games. The method used applies several concepts, e.g., prefix trees, to extract knowledge from the vast amount of data.

1 Introduction

Go is a strategic two player game, played on a 19×19 board. For the rules we refer to [7]. Many studies have been done with regard to go in general, cf. [6, 8], and to joseki, localized exchanges of stones that are considered fair for both players. We will give an algorithm that finds and catalogues as many joseki as it can, as well as the global circumstances under which they are likely to be played, by analyzing a large number of professional go games.

The algorithm is able to acquire knowledge out of several complex examples of professional game play. As such, it can be seen as data mining [10], and more in particular sequence mining, e.g., [1]. The use of prefix trees in combination with board positions seems to have a lot of potential for the game of go.

In Section 2 we explain what joseki are and how we plan to find them. Section 3 will explain the algorithm in more detail using an example game from a well-known database [2]. In Section 4 we mention some issues concerned with symmetry. Section 5 will discuss the results of the algorithm. We try to explore the global circumstances under which a joseki is played in Section 6. Section 7 contains conclusions and discusses further research.

Michiel Helvensteijn
LIACS, Leiden University, The Netherlands, e-mail: mhelvens@liacs.nl

Please use the following format when citing this chapter:

Helvensteijn, M., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 8796.

2 Joseki

There are several definitions of *joseki*, see, e.g., [5] and [4]. We will use a somewhat adapted definition, which uses elements from other sources:

> A joseki is a localized sequence of play in the game of go, in which both players play locally optimal moves. These sequences occur often in recorded games, especially in the corners of the board and the beginning of the game.

It is an important property of a joseki that it is a local phenomenon. It takes place in a certain part of the board and moves that are played elsewhere (before, during or after the joseki) are not part of it. The players can sometimes break away from a joseki and get back to it later. This way, multiple joseki can be in progress at the same time. The move that breaks away from a joseki is called a *tenuki*. The move that continues the joseki after a tenuki is called a *follow-up play*.

We do not think that a joseki results in a fair outcome for both players by definition, as is often stated in other definitions. In fact, joseki do not result in a fair outcome if played under the wrong global conditions. Of course, if a sequence did not result in a fair outcome under *some* global condition, it could never have been played often enough to be noticed and given the name joseki. This is important. Professional players do not blindly play joseki in their games, not even under optimal global conditions. At most, they use their knowledge of joseki as a heuristic. They play the move that they think is best, and that is how joseki are found. This is why we have not mentioned fairness in the definition, it is implied already. It is also irrelevant to the algorithm. A computer can not calculate whether a sequence is fair. Instead we choose to rely on human intuition, in that a sequence must be a joseki under the above definition if it is played often enough by professionals.

3 The algorithm

In this section we give an algorithm to find joseki in a given database. We will use a database from the Go4Go website [2], which contains 13,325 go games played by professional go players. It is the job of the algorithm to analyze the games from this database and eventually output the joseki (plural) that were found. The algorithm is depicted in Figure 1 and Figure 2, which show phase 1 and 2 of the algorithm respectively. In this section we mention some symmetry related issues; however, the formal treatment of this subject is postponed until Section 4.

The first phase extracts all distinguishable sequences from the games in the database and stores them in a prefix tree. The second phase prunes that tree so that only the more interesting sequences remain, resulting in a tree of joseki.

3.1 Phase 1

At the end of this phase, we will have a tree of all distinct move-sequences[1] to be found in the games. Phase 1, step 1 is basically a nested loop that iterates over all moves of all games in the database. Every move is compared to all currently stored sequences that belong to the current game. Its Manhattan distance[2] to the stones of those sequences is used to determine whether it belongs to one of them. It is also possible for one move to belong to more than one sequence.

Because the joseki we are looking for are played mostly in the corners and in the beginning of the game, the algorithm will stop looking for stones after each corner contains at least 20 stones. This means that at the least, we will examine 80 moves. Anything more than that means we have entered mid-game. The reason we don't just look at the first 80 stones instead is that sometimes a single corner can remain empty for a large portion of the game, which means we might miss some obvious joseki.

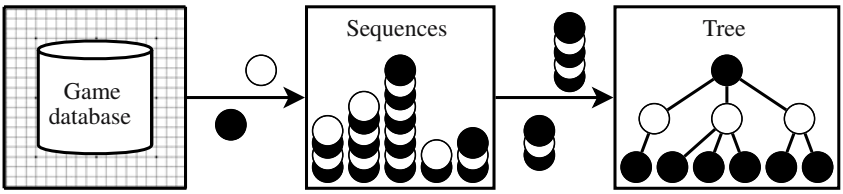


Fig. 1 The algorithm, phase 1: creating the tree

Step 2 moves these sequences to the tree, after a possible transformation (see Section 4). It is implemented as a prefix tree, a structure very popular in current data mining algorithms [3]. The root of this tree represents the empty board. Its children are the first moves of the sequences, and so on. Each node in the tree represents a sequence prefix to that point and its children represent its continuations. Each node contains a mapping of point→node (where "point" is a point on the board or a tenuki) to find its children. This provides very fast lookup and insertion of sequences. Each node also has a counter indicating how often a certain sequence has been played. An insertion increases the right counters in the tree and adds new nodes if necessary.

For efficiency, step 1 and step 2 are performed simultaneously. After every game, the sequences are added to the tree and the sequence storage is made empty.

[1] A sequence is a potential joseki. It is also called a sequence because it might eventually turn out to be irrelevant and be pruned from the tree in phase 2. Only the sequences that survive this process are called joseki.

[2] The Manhattan distance between two points is the absolute difference between their x-coordinates plus the absolute difference between their y-coordinates: the Manhattan distance between the points (x_1, y_1) and (x_2, y_2) is $|x_1 - x_2| + |y_1 - y_2|$.

3.2 Phase 2

Phase 2 consists of pruning and printing (in SGF format [9]) the tree that we built in phase 1. It removes the irrelevant sequences with a pruning function, that accepts or rejects a sequence based on its *pruning score*, i.e., its frequency in the prefix tree.

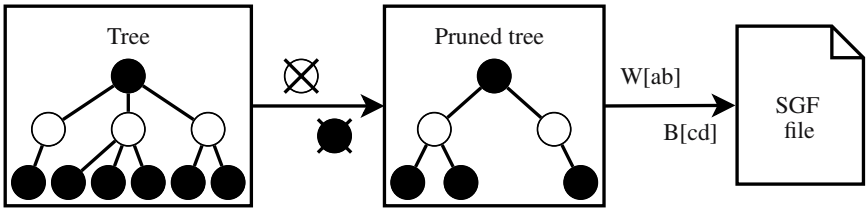


Fig. 2 The algorithm, phase 2: pruning the tree

Because of the nature of the prefix tree, the counter of any node is greater than or equal to the sum of the counters of its children, and so in particular greater than or equal to the counter of any child node. The basic pruning approach is to cut off any subtree that does not have a counter greater than or equal to a given threshold value. We have experimented with several threshold values. The optimal value appears to be around 1% of the amount of games in the database.

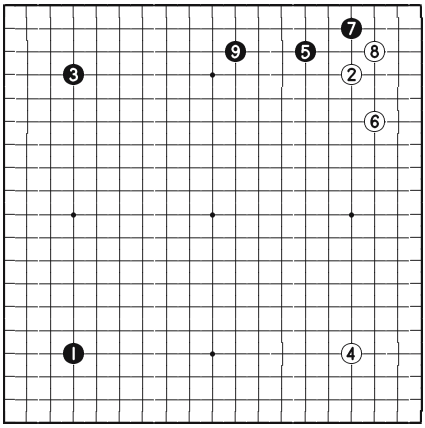
3.3 Example

We clarify the algorithm through an example game, a match between two strong players, Xie He (white) and Duan Rong (black), from the first round of the 18th Chinese CCTV Cup. See Diagram 1.

Both players first occupy the four corner star points. As can be seen in Diagram 1, each of the first four moves starts a new sequence. For now we will call them sequence 1, 2, 3 and 4.

Black 5 starts the first joseki of the match. It ends with black 9. A move belongs to a sequence if it is within Manhattan distance x of it, where x is a predefined threshold: the *sequence binding distance*. For this example, $x = 5$. Black 5 is clearly only within proximity of white 2, and so it is added to sequence 2. The same holds for white 6 to white 8. Black 9 also belongs to sequence 2, because it is within proximity of black 5, which was added to the sequence earlier.

White 10, black 11 and white 12 are added to sequence 3 (see Diagram 2). One will notice that black 11 is only just outside the reach of sequence 2. Black 13 starts a long joseki that ends with black 23. All of those moves are part of sequence 4.



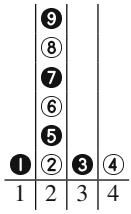
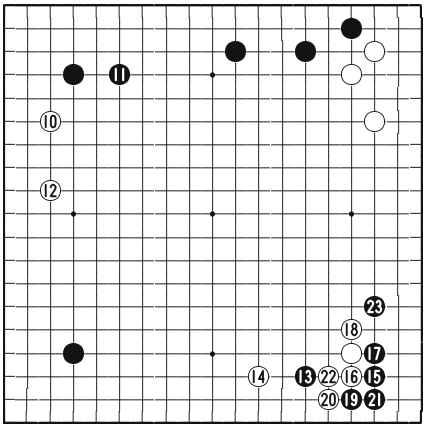


Diagram 1 Xie He vs. Duan Rong, part 1



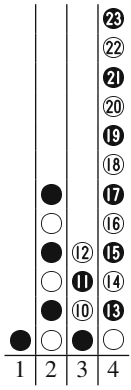
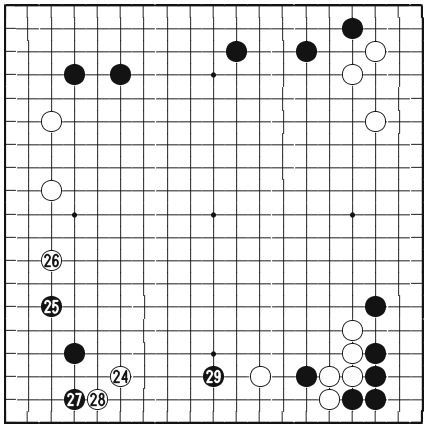


Diagram 2 Xie He vs. Duan Rong, part 2

Such long, isolated joseki are not as uncommon as one might imagine, as proved by this algorithm. That exact joseki is found 566 times in the database.

White 24 and black 25 add another small joseki to the collection, in sequence 1 (Diagram 3). But something else has also happened here. Black 25 is within range of white 12, as well as black 1, so it is also part of sequence 3, as is white 26. After two non-joseki moves, black 29 does the same thing. It is part of both sequence 1 and 4. This is bound to happen as more and more stones are played on the board. But the assumption is that either joseki are played in isolation before the sequences start interfering with each other or that only one of the sequences really "owns" the new move. It is not unthinkable that a stone with a distance of 5 from a sequence doesn't really belong to it. For example, if black 25 were part of any joseki, it would



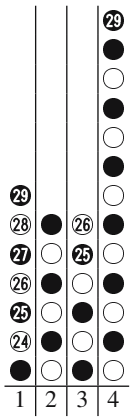


Diagram 3 Xie He vs. Duan Rong, part 3

be sequence 1. These things will be recognized in phase 2, when follow-up moves that do not belong to the joseki are pruned out of the tree.

We have now discovered and completed all joseki the algorithm is able to find in this example game. However, the algorithm will not know it at this point and will continue to build the sequences until at least 20 stones have been played in each corner.

Prefixes of all four of the sequences played so far will turn out to be important joseki in the final tree. Table 1 shows these joseki.

Table 1 Xie He vs. Duan Rong, four joseki

	Joseki	Transformation	Color-swap?
1	❶ ㉔ ㉕	**H**	No
2	② ❺ ⑥ ❼ ⑧ ❾	**V**	Yes
3	❸ ⑩ ⓫	**D**	No
4	④ ❸ ⑭ ❺ ⑯ ❼ ⑱ ❾ ⑳ ❶ ㉒ ❸	**H** × **V**	Yes

The first column gives the sequence reference number. The "Joseki prefix" column gives the prefix of the sequence that forms the joseki. In other words, the stones that would be pruned in phase 2 are not shown here. The "Transformation" column shows the matrix manipulation that should be applied to each move of the sequence, so it will yield the sequence's normal form (see Section 4). Here **H** means a reflection in the horizontal ($y = 10$) axis; **V** means a reflection in the vertical ($x = 10$) axis; **D** means a reflection in the diagonal axis on which black 3 and white 4 are played; and $\times$ is the matrix multiplication operator. So sequence 4 has to be reflected in the horizontal and vertical axes to get to its normal form. The "Color-swap?" column

indicates if the colors of the stones need to be swapped from black to white or the other way around. This is the case for all sequences where white moves first, because by convention, black has the first move. We will adopt this convention to get our normal form.

This procedure is performed for all games in the database, resulting in a tree with still many irrelevant sequences. After phase 2, however, it will be a relatively small tree with assorted recognizable joseki.

4 Symmetry

The board has a symmetry group (the dihedral group D_4) with 8 elements, which can be generated by a rotation by 90° and a reflection in one of the four board axes (the horizontal, vertical and two diagonal axes). Reflecting twice is a rotation around the intersection of the axes, i.e., the board center or *tengen*. Another dimension of symmetry with regard to joseki is color, in that two spatially identical stones are still equivalent, even if one is black and the other is white.

This symmetry can be extended to a sequence of moves. When two joseki are equivalent in this way, we want the algorithm to recognize this. So when it transfers a sequence of moves to the tree, it first transforms each of them using reflections and color-swaps such that the resulting sequence will start with a black move in a designated triangular space on the board. Note that it is often necessary to consider more than just the first move in order to determine the exact transformation.

In theory, another transformation is possible: translation. Joseki that occur along the top edge of the board may be equally valid two places to the right or to the left. The algorithm does not take this into account, however, because this validity can be extremely complicated to judge. It is also very situation dependent, unlike reflections, rotations and color-swaps which are always valid.

5 Results

In this section we mention the results of experiments on the database [2], consisting of 13,325 games. We have experimented with several parameter settings. Each run took only a few seconds, which is not surprising in view of the linear nature of the algorithm. It was found that the following settings gave the best results:

pruning score: 150
sequence binding distance: 5

The resulting tree (Figure 3) contains 81 leafs, meaning 81 complete joseki. However, the algorithm does not only find joseki, but also a lot of what might more properly be called *fuseki* structures (opening game sequences). This is not surprising, since the algorithm looks primarily in the corners of the board and the opening

game, which are exactly the time and place fuseki structures are formed. The set of joseki and the set of fuseki seem to overlap when one only considers the opening game. The resulting tree shows some well-known joseki and fuseki.

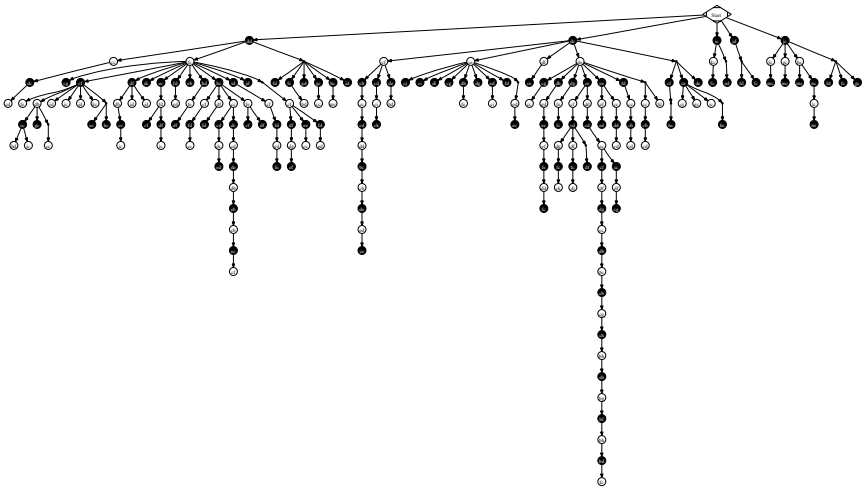


Fig. 3 Joseki tree, pruning score = 150, sequence binding distance = 5

6 Global influence

More is still to be known about these joseki. We know now which joseki (and fuseki) are played, but we still do not know *when* they are played. As explained in Section 2, this is important information.

There are many factors that could have an influence on the "fairness" or "validity" of a joseki, like ladder breakers, nearby influential stones and the overall score in points (a player who is ahead will most likely play a more balanced game). But another important factor is global influence around the board.

The algorithm calculates the global influence around a joseki. This information extends the output of the algorithm, but does not alter it. The algorithm as explained in Section 3 remains mostly unchanged, though the current state of the game is always kept in memory, so global influence can be investigated. And influence direction and color is of course transformed along with the sequence before being put into the tree.

Diagram 4 shows how this influence is calculated, using a new example. The stone marked △ has just been played and added to the bottom-right sequence. The influence reaching this sequence has to be calculated for that move. From the left edge of the sequence's bounding-box, a search to the left is done for each row. Each

row is scored (17 minus the distance from the bounding-box to the first stone, capped at 0)[3]. Certain threshold values can determine if the total score (all row-scores added together) is white, black or neutral. The net score for this particular search turns out to be 15 for black. This means black has the most influence in that direction, which is quite clearly the case. This procedure can be repeated for the other three sides, though two of them almost always have zero influence, since most joseki are played in a corner.

This influence information in the tree is stored in aggregate, and so it is determined what the most likely global circumstances are for each stage of each joseki. Manual inspection of the tree indicates that for most joseki, influence is most definitely a factor, as expected.

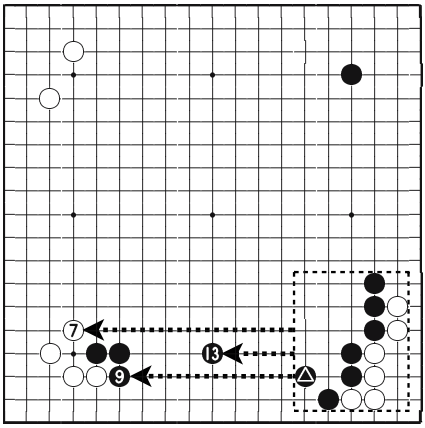


Diagram 4 Calculation of left-side influence

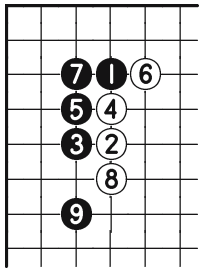


Diagram 5 The nadare; most variations branch off after white 6

7 Conclusions and future research

The technique of finding joseki in a database as described in this paper certainly has merit. It finds some well-known joseki and fuseki sequences and none of them seem out of place. The search for global influence also produced promising results.

For example, the algorithm finds the joseki shown in Diagram 5. This joseki is known as the *nadare*. This joseki, and some common variations on it, are described in the Kosugi/Davies book [5]. In the tree of Figure 3 it is the joseki of 9 moves deep, closest to the really long one (which seems to be a variation on the nadare

[3] If a stone is closer to the sequence, it has more influence on it, making it more likely that a player will deviate from the joseki that would have been played without this influence. Even if a stone is on the other side of the board, though, it can have influence. The number 17 was chosen experimentally as the furthest distance from which a stone could still have *any* influence.

not described by Kosugi and Davies). By far most of the occurrences of this joseki have a consistent black influence from below throughout the sequence. To a lesser degree, white seems to have more influence to the right, which would play well with white's new wall. The fact that verifiable joseki such as this one can be found like this is very encouraging.

Because of the high ranking of the players in the database, not a big subset of the known joseki is found (there are thousands). It might be interesting to try this algorithm on a database of weaker players.

In the algorithm, the decision whether a move belongs to a sequence or not is decided by Manhattan distance. Other distance-measures could be used instead, and might be more appropriate. And the tree is now pruned with a single strict pruning-score. It may be advisable, in future research, to explore other possibilities.

Acknowledgements The author would like to thank Hendrik Blockeel, Walter Kosters and Jan Ramon for their help with this project.

References

1. Gouda, K., Hassaan, M. and Zaki, M. J.: PRISM: A Prime-Encoding Approach for Frequent Sequence Mining, Proceedings of the 7th IEEE International Conference on Data Mining, pp. 487–492 (2007)
2. Go4Go.net [online] http://www.go4go.net/v2
3. Han, J., Pei, J. and Yin, Y.: Mining Frequent Patterns without Candidate Generation, Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 1–12 (2000)
4. Joseki, Wikipedia [online] http://en.wikipedia.org/wiki/Joseki
5. Kosugi, K. and Davies, J.: Elementary Go Series, Volume 2, 38 Basic Josekis, Kiseido Publishing Company, Eighth Printing, 2007
6. Ramon, J. and Blockeel, H.: A Survey of the Application of Machine Learning to the Game of Go, Proceedings of the First International Conference on Baduk (Sang-Dae Hahn, ed.), pp. 1–10 (2001)
7. Sensei's Library, The Collaborative Go Website [online] http://senseis.xmp.net
8. Silver, D., Sutton, R. and Müller, M.: Reinforcement Learning of Local Shape in the Game of Go, Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI), pp. 1053–1058 (2007)
9. Smart Game Format Specifications [online] http://www.red-bean.com/sgf
10. Tan, P. N., Steinbach, M. and Kumar, V.: Introduction to Data Mining, Addison-Wesley, 2006

A Fuzzy Semi-Supervised Support Vector Machines Approach to Hypertext Categorization

Houda Benbrahim[1] and Max Bramer[2]

Abstract Hypertext/text domains are characterized by several tens or hundreds of thousands of features. This represents a challenge for supervised learning algorithms which have to learn accurate classifiers using a small set of available training examples. In this paper, a fuzzy semi-supervised support vector machines (FSS-SVM) algorithm is proposed. It tries to overcome the need for a large labelled training set. For this, it uses both labelled and unlabelled data for training. It also modulates the effect of the unlabelled data in the learning process. Empirical evaluations with two real-world hypertext datasets showed that, by additionally using unlabelled data, FSS-SVM requires less labelled training data than its supervised version, support vector machines, to achieve the same level of classification performance. Also, the incorporated fuzzy membership values of the unlabelled training patterns in the learning process have positively influenced the classification performance in comparison with its crisp variant.

1 Introduction

In the last two decades, supervised learning algorithms have been extensively studied to produce text classifiers from a set of training documents. The field is considered to be mature as an acceptable high classification effectiveness plateau has been reached [1]. It has become difficult to detect statistically significant differences in overall performance among several of the better systems even though they are based on different technologies.

However, to achieve these good results, a large number of labelled documents is needed. This coincides with the conclusions from computational learning theory that state that the number of training examples should be at least a multiple of the number of features if reasonable results are sought [2]. Often, several thousand features are used to represent texts, and this leads to a need for thousands of labelled training documents. Unfortunately, obtaining this large set is a difficult task. Labelling is usually done using human expertise, which is tedious,

[1] Dr. Houda Benbrahim

University of Portsmouth, School of Computing, PO1 3HE, UK. email: houda.benbrahim@port.ac.uk

[2] Prof. Max Bramer

University of Portsmouth, School of Computing, PO1 3HE, UK. email: max.bramer@port.ac.uk

Please use the following format when citing this chapter:

Benbrahim, H. and Bramer, M., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 97406.

expensive, time consuming and error prone. On the other hand, unlabelled documents are often readily available in large quantities, and one might prefer to use unsupervised learning algorithms (restricted here to clustering). Yet, learning solely from unlabelled documents cannot be used to classify new documents into predefined classes because knowledge about classes is missing. In this case, semi-supervised learning comes to the rescue as it lies in between supervised and unsupervised learning approaches. It takes advantage of the strengths of both learning paradigms, i.e. it learns accurate classifiers and exploits the unlabelled data, and discards their major drawbacks, i.e. their need for a large labelled training set and their inability to identify the classes.

The principal question that may arise in semi-supervised learning is how to combine labelled and unlabelled data in the learning system. In order to benefit from unlabelled data in a supervised learning model, a learner must augment unlabelled examples by class labels in some way. However, fully using this newly labelled and originally unlabelled set of training documents in the supervised learning process may harm the performance of the resulting classifier.

Classifying the unlabeled data using any classifier is error prone. Consequently, the newly labelled data imputed in the training set might be noisy, and this usually harms the performance of the learning algorithm as its performance might decrease with noisy training data. A possible solution to this problem is to modulate the influence of the originally unlabelled data in the supervised training phase. This might be achieved by introducing fuzzy memberships to unlabelled documents. In this case, a fuzzy membership value is associated with each document such that different documents can have different effects in the learning of the classifier.

In this paper, a Fuzzy Semi-Supervised Support Vector Machine approach is proposed for hypertext categorization.

Many researchers have studied semi-supervised support vector machines, which attempt to maximize the margin on both labelled and unlabelled data, by assigning unlabelled data to appropriate classes such that the resulting margin is the maximum. Earlier works include Transductive support vector machine (TSVM) first introduced by [3], which uses the unlabelled test set in the training stage. The problem with TSVM is that its training is more difficult. [4] uses an iterative method with one SVM training on each step, while mixed integer programming was used in S3VM [5]. [6] formulated the problem as a concave minimization problem which is solved by a successive linear approximation algorithm and produced V3SVM and CV3SVM.

SVM is sensitive to noise and outliers in the training dataset [7]. To solve this problem, one approach is to do some processing on the training data to remove noise or outliers, and use the remaining set to learn the decision function [8]. Among the other approaches is the introduction of fuzzy memberships to data points such that different data points can have different effects in the learning of the separating hyperplane. Few fuzzy support vector machine approaches exist that treat noise and outliers as less important and let these points have lower membership values [9, 10].

This paper deals with a proposed Fuzzy-Semi-Supervised Support Vector machine framework. It is introduced in two steps. First, we describe the concept of semi-supervised clustering guided by labelled data. Then, we define how unlabelled data is partially incorporated into the learning process of the support vector machines model. Several experiments will be conducted to provide empirical evidence about (i) the effect of the number of labelled training documents in the fuzzy semi-supervised support vector machines learning process, and (ii) the effect of the number of unlabelled training documents in the fuzzy semi-supervised support vector machines learning process.
Fuzzy semi-supervised support vector machines approach is described in section 2. Section 3 presents experiments and results, comparing different classification algorithms. Section 4 concludes the paper.

2 Fuzzy Semi-Supervised Support Vector Machines Approach

Semi-supervised learning is halfway between supervised and unsupervised learning. In addition to unlabelled data, the algorithm is also provided with labelled data. In this case, the data set X can be divided into two parts: set $X_L = \{x_1, \ldots, x_L\}$, for which labels $Y_L = \{y_1, \ldots, y_L\}$ are provided, and a set $X_u = \{x_1, \ldots, x_u\}$ where the labels are not known. The objective of semi-supervised learning is to benefit from both supervised and unsupervised learning when combining labelled and unlabelled data.
The open question that may arise is how to take advantage of the unlabelled data to build a classifier. There are many approaches to this problem. The one adopted in this work is to train a classifier based on labelled data as well as unlabelled data. Typically, the unlabelled data is clustered then labelled, and then the augmented labelled data is used to train the final classifier. Two key issues in this approach are (i) how to impute labels to unlabelled data and (ii) how to use the augmented labelled data to train the classifier.
The semi-supervised task in this paper can be formulated as follows: As a first step, a clustering algorithm (unsupervised learning) can be applied to discover groups in the unlabelled data; in this case, a c-means clustering algorithm [11] might be used. However, determining a suitable number of clusters and generating a suitable starting solution is a challenge for clustering algorithms. To overcome this dilemma, labelled data can be used in the unsupervised learning step. Therefore, a semi-supervised c-means algorithm [12] is applied. It also allows labelling the discovered clusters/groups. As a second step, a model is learned based on a supervised learning algorithm namely support vector machines trained by the whole set of labelled data and the newly labelled unlabelled data.
In the crisp support vector machines approach, each training pattern has the same weight/importance in deciding about the optimal hyperplane. In this paper, and in this proposed FSS-SVM algorithm, the originally unlabelled data along with their imputed class labels in addition to the labelled data are used as a training set.

However, classical SVM learning is sensitive to noisy data because of the inherent "over-fitting" problem. This may increase the classification error [7, 13], and in order to decrease the effect of this possible noise that might originate from the unlabelled training sample, each training pattern is assigned a membership value, that corresponds to its weight in SS-FCM, to modulate the effect of the training data on the learning process of SVM. FSS-SVM also maximizes the margin of separation and minimizes the classification error so that a good generalization can be achieved. To reach that objective, FSS-SVM models the effect of unlabelled data incorporated in the training set.

FSS-SVM

The proposed fuzzy semi-supervised support vector machines algorithm works as follow:

- Let X be the set of training examples. X is divided into two parts: set $X_L = \{x_1, \ldots, x_L\}$, for which labels $Y_l = \{y_1, \ldots, y_L\}$ are provided, and a set $X_u = \{x_1, \ldots, x_u\}$ where the labels are not known.

- SSFCM is used to impute the class labels of the unlabelled data set. Each unlabelled example x_j is assigned to
  class $y_j^u = \arg\max_{i \in \{1,\ldots,c\}} u_{ij}^u, \forall j \in \{1,\ldots,n_u\}$ with membership value μ_{ij}.

- The set $X_L = \{(x_1, y_1), \ldots (x_L, y_L)\}$ of labelled patterns, and a set of $X_u = \{(x_1, y_1, \mu_1), \ldots (x_u, y_u, \mu_u)\}$ of unlabelled patterns with their corresponding imputed class label and fuzzy membership value in that class are used as a training set for FSS-SVM.

- The optimal hyperplane problem can be regarded as the solution to:

$$\min \tfrac{1}{2}\|w\|^2 + C\left[\sum_{i=1}^{L}\xi_i + \sum_{j=1}^{u}\mu_j\xi_j^*\right]$$

Subject to:
$$y_i\left[\langle w, x_i\rangle + b\right] \geq 1 - \xi_i, i = 1,\ldots,L$$
$$y_j\left[\langle w, x_j\rangle + b\right] \geq 1 - \xi_j^*, j = 1,\ldots,u$$
$$\xi_i \geq 0, i = 1,\ldots,L$$
$$\xi_j^* \geq 0, j = 1,\ldots,u$$

Since ξ_i is the measure of error of a pattern x_i in the SVM learning process, the term $\mu_i\xi_i$ is then the measure of error with different weighting. The smaller the value μ_i , the smaller the effect of ξ_i , which means that the corresponding x_i is treated as less important.
Hence the solution is:

$$\lambda^* = \arg\min_{\lambda} \tfrac{1}{2} \sum_{i=1}^{L+u} \sum_{j=1}^{L+u} \lambda_i \lambda_j y_i y_j \left\langle x_i, x_j \right\rangle - \sum_{i=1}^{L+u} \lambda_k$$

With constraints:

$$0 \le \lambda_i \le C, i = 1,..., L$$

$$0 \le \lambda_i \le \mu_i C, i = 1,..., u$$

$$\sum_{j=1}^{L+u} \lambda_j y_j = 0$$

3 Experiments

In this section, several experiments have been conducted to provide empirical evidence that learning from both labelled and unlabelled data through our proposed fuzzy semi-supervised support vector machines approach outperforms the traditional crisp supervised SVM learning algorithm which learns only from labelled data.

Mainly, we will check in those experiments:

- The effect of the number of labelled training documents in the fuzzy semi-supervised support vector machines learning process.
- The effect of the number of unlabelled training documents in the fuzzy semi-supervised support vector machines learning process.

3.1 Datasets

BankSearch [14] and Web->KB (www.cs.cmu.edu/~webkb/) hypertext datasets were used to evaluate the performance of the new classifier. However, we do not have available unlabelled data related to these datasets. For this reason, 30% of the available data was held aside and used as unlabelled data.

3.2 The classification task

The classification problem for both datasets is a single-label-per-document multiclass case, which means that the classifiers must decide between several categories, and each document is assigned to exactly one category. However, all the classification tasks were mapped into their equivalent binary classification problems. The one against all method was used to split the n-class classification problem into n-binary problems.

3.3 Document presentation

The pre-processing step for documents in both datasets comprises the following. The content of HTML pages, along with their corresponding extra information extracted. Each document representation is enhanced by its title + link anchor + meta data + similar neighbour [15]. However, when using labelled and unlabelled data for learning a classifier, we have to specify how the unlabelled data will participate in the different steps of the hypertext representation, namely, indexation, feature reduction and vocabulary generation.
For the indexation phase, all indexes occurring in both labelled and unlabelled documents are taken into consideration; this is to enrich the vocabulary of the dataset in case there are a small number of labelled documents.
Dimensionality reduction can also be applied when dealing with labelled and unlabelled documents. However, some restrictions are posed. For example, the information gain feature selection technique cannot be used in this case as it requires that the class label be known. To be able to use it anyway, the measure can be restricted to labelled documents, this leads to loss of information related to unlabelled data. Moreover, this class-dependent feature selection tends to be statistically unreliable as we are assuming that the labelled documents are scarce.
Hence, for feature reduction, we apply only stop word removal, stemming, and elimination of words that occurs at most once in the training dataset. Then all the remaining indexes are used to build the dictionary.

3.4 Evaluation procedure

Two different evaluation procedures were carried out for the two datasets.
For WEB->KB dataset, a 4-fold leave-one-university-out-cross-validation was used. That is for each experiment, we combined the examples of three universities to learn a classifier which was then tested on the data of the fourth university.
For the BankSearch dataset, the holdout method is used. The dataset is randomly split into 70% training and 30% testing and repeated 30 times.
Micro-averaged F1 and accuracy measures were used to evaluate the classifiers.

3.5 The effect of the number of labelled training documents

Figures 1 and 2 show the classification F1 measure of the fuzzy semi-supervised support vector machines (FSS-SVM) on the two hypertext datasets when the number of labelled training documents is varied, and the number of unlabelled training documents is kept fixed (30% from each class). The results are contrasted with the learning results of SVM (which learns from only the labelled training documents), SSFCM and SS-SVM.

SS-SVM is a simple version of a semi-supervised SVM. The originally unlabelled data is classified using SSFCM algorithm. Then, each pattern is crisply assigned to the class that corresponds to the highest value in the resulting membership matrix.
The horizontal axes indicate the number of labelled training documents. For instance, a total of 11 training documents for the BankSearch dataset correspond to 1 (one) document per class and a total of 40 training documents correspond to10 documents per class for the Web->KB dataset. The vertical axes indicate the F1 measure on the test sets.
In all experiments, the fuzzy semi-supervised support machine performs better than its supervised version when the number of labelled training documents is small, i.e. FSS-SVM can achieve a specific level of classification accuracy with much less labelled training data. For example, with only 550 labelled training examples for the BankSearch dataset (50 documents per class), FSS-SVM reaches 0.65 F1 measure classification, while the traditional SVM classifier achieves only 0.5. For the same labelled training set size, F1 measure of SS-SVM is 0.46 and 0.55 for SSFCM. In other words, to reach 0.65 classification F1 measure, for example, SVM requires about 1100 and FSS-SVM only 550 labelled training documents.
Similarly, for the WebKB dataset, the performance increase is smaller but substantial; this may be because of the small size of the unlabelled data. For instance, for 80 labelled training examples (20 documents per class), SVM obtains 0.29 F1 measure and FSS-SVM 0.59, reducing classification error by 0.3. For the same number of labelled documents, SS-SVM achieves 0.36 F1 measure and SSFCM 0.43.
For both datasets, FSS-SVM is superior to SVM when the amount of labelled training data is small. The performance gain achieved by the semi-supervised learners decreases as the number of labelled training documents increases. The reason for this is that more accurate classifiers can be learned from the labelled data alone. As the accuracy obtained through plain supervised learning approaches a dataset-specific plateau, we barely benefit from incorporating unlabelled documents through semi-supervised learning.

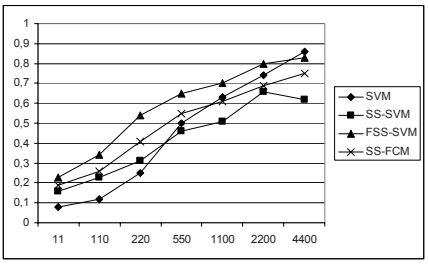


Figure 1: Classifiers F1 measure with different numbers of labelled data used for training for BankSearch dataset.

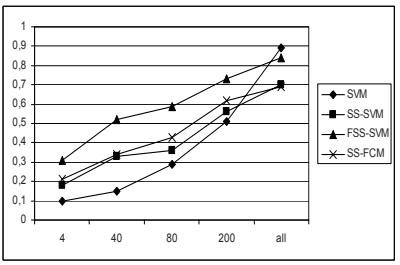


Figure 2: Classifiers F1 measure for different numbers of labelled data for training for WEB->KB dataset.

In fact, note that the accuracy of FSS-SVM also degrades when the number of labelled training documents is very large. For instance, with 4400 labelled training examples (400 documents per class) on the BankSearch dataset, classification F1 measure decreases from 0.86 to 0.81.
To summarize, the results for the fuzzy semi-supervised support vector machines classifier show that the benefit we may achieve from the use of unlabelled documents strongly depends on the number of labelled training documents. The learning performance increases as the number of labelled training documents increases. So, when the number of labelled training documents is small, the learning algorithm needs more help. Therefore, the learner benefits from the additional unlabelled documents even though their imputed class labels are uncertain. However, it seems that the fuzzy hyperplane margin that modulates the influence of the imputed labelled data enhances the classifier's performance. SSSVM performance degrades in some cases in comparison with that of SVM as more unlabelled documents are incorporated in the training set. This might be explained by the fact that the imputed labels of the unlabelled data tend to be incorrect as they are predicted by SSFCM, and therefore may not be correctly classified.

3.6 The effect of the number of unlabelled training documents

In the previous set of experiments, we have shown that the extent to which we may benefit from unlabelled documents depends on the number of labelled training documents available. Obviously, this benefit will also depend on the number of unlabelled documents. The results below examine the effect of the unlabelled set size on the classifier's performance. Figures 3 and 4 show the classification F1 measure of FSS-SVM with different numbers of labelled training documents on the BankSearch and WEB->KB datasets when the number of unlabelled documents is varied (10%, 20% or 30% of the available unlabelled data). In all cases, adding unlabelled data often helps learning more effective classifiers. Generally, performance gain increases as the amount of labelled data decreases. Also, performance gain increases with the number of unlabelled documents until it reaches a plateau.

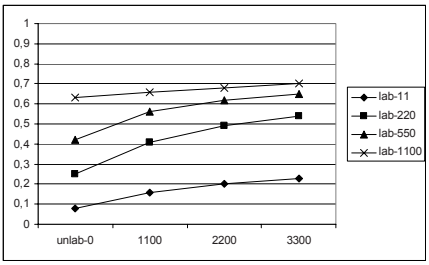


Figure 3: FSS-SVM F1 measure with different numbers of labelled training documents and different numbers of unlabelled training documents for BankSearch dataset.

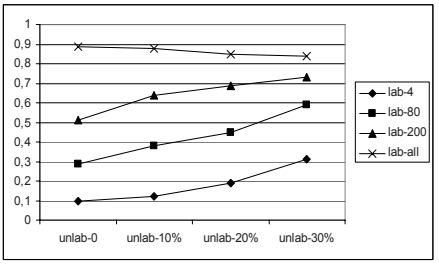


Figure 4: FSS-SVM F1 measure with different numbers of labelled training data and different numbers of unlabelled training data for WEB->KB dataset.

4 Conclusion

In this paper, we have presented a fuzzy semi-supervised support vector machines learning approach to hypertext categorization. This is learning from labelled and unlabelled documents. This is a crucial issue when hand labelling documents is expensive, but unlabelled documents are readily available in large quantities, as is often the case for text classification tasks. The following summarizes the results of the empirical evaluation:

• FSS-SVM can be used to learn accurate classifiers from a large set of unlabelled data in addition to a small set of labelled training documents. It also outperforms its supervised version (SVM). In other words, FSS-SVM requires less labelled training data to achieve the same level of classification effectiveness.

References

[1] Liere, R. and P. Tadepalli (1996). “The use of active learning in text categorization.” Proceedings of the AAAI Symposium on Machine Learning in Information Access.

[2] Lewis, D. D. (1992). “Feature selection and feature extraction for text categorization.” Proceedings of the workshop on Speech and Natural Language: 212-217.

[3] Vapnik, V. N. (1998). Statistical learning theory, Wiley New York.

[4] Joachims, T. (1999). “Transductive inference for text classification using support vector machines.” Proceedings of the Sixteenth International Conference on Machine Learning: 200-209.

[5] Bennett, K. and A. Demiriz (1998). “Semi-supervised support vector machines.” Advances in Neural Information Processing Systems **11**: 368-374.

[6] Fung, G. and O. Mangasarian (1999). “semi-supervised support vector machines for unlabeled data classification.” (Technical Report 99-05). Data mining Institute, University of Wisconsin at Madison, Madison, WI.

[7] Zhang, X. (1999). “Using class-center vectors to build support vector machines.” Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop: 3-11.

[8] Cao, L. J., H. P. Lee, et al. (2003). “Modified support vector novelty detector using training data with outliers.” Pattern Recognition Letters **24**(14): 2479-2487.

[9] Lin, C. F. and S. D. Wang (2002). “Fuzzy support vector machines.” IEEE Transactions on Neural Networks **13**(2): 464-471.

[10] Sheng-de Wang, C. L. (2003). “Training algorithms for fuzzy support vector machines with noisy data.” Neural Networks for Signal Processing, 2003. NNSP'03. 2003 IEEE 13th Workshop on: 517-526.

[11] Bezdek, J. C. (1981). Pattern Recognition with Fuzzy Objective Function Algorithms, Kluwer Academic Publishers Norwell, MA, USA.

[12] Bensaid, A. M., L. O. Hall, et al. (1996). “Partially supervised clustering for image segmentation.” Pattern Recognition 29(5), 859-871.

[13] Guyon, I., N. Matic, et al. (1996). “Discovering informative patterns and data cleaning.” Advances in knowledge discovery and data mining table of contents: 181-203.

[14] Sinka, M. P. and D. W. Corne (2002). “A large benchmark dataset for web document clustering.” Soft Computing Systems: Design, Management and Applications **87**: 881-890

[15] Benbrahim, H. and M. Bramer (2004). “Neighbourhood Exploitation in Hypertext Categorization.” In Proceedings of the Twenty-fourth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, Cambridge, December 2004, pp. 258-268. ISBN 1-85233-907-1

NEURAL NETWORKS

Estimation of Neural Network Parameters for Wheat Yield Prediction

Georg Ruß, Rudolf Kruse, Martin Schneider, and Peter Wagner

Abstract Precision agriculture (PA) and information technology (IT) are closely interwoven. The former usually refers to the application of nowadays' technology to agriculture. Due to the use of sensors and GPS technology, in today's agriculture many data are collected. Making use of those data via IT often leads to dramatic improvements in efficiency. For this purpose, the challenge is to change these raw data into useful information. This paper deals with suitable modeling techniques for those agricultural data where the objective is to uncover the existing patterns. In particular, the use of feed-forward backpropagation neural networks will be evaluated and suitable parameters will be estimated. In consequence, yield prediction is enabled based on cheaply available site data. Based on this prediction, economic or environmental optimization of, e.g., fertilization can be carried out.

1 Introduction

Due to the rapidly advancing technology in the last few decades, more and more of our everyday life has been changed by information technology. Information access, once cumbersome and slow, has been turned into "information at your fingertips" at high speed. Technological breakthroughs have been made in industry and services as well as in agriculture. Mostly due to the increased use of modern GPS technology and advancing sensor technology in agriculture, the term *precision agriculture* has been coined. It can be seen as a major step from uniform, large-scale cultivation of soil towards small-field, precise planning of, e.g., fertilizer or pesticide usage. With the ever-increasing

Georg Ruß, Rudolf Kruse
Otto-von-Guericke-Univ. of Magdeburg, e-mail: {russ,kruse}@iws.cs.uni-magdeburg.de

Martin Schneider, Peter Wagner
Martin-Luther-Univ. of Halle, e-mail: {martin.schneider,peter.wagner}@landw.uni-halle.de

Please use the following format when citing this chapter:

RußG., Kruse, R., Schneider, M. and Wagner, P., 2008, in IFIP In ternational Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 109418.

amount of sensors and information about their soil, farmers are not only harvesting, e.g., potatoes or grain, but also harvesting large amounts of data. These data should be used for optimization, i.e. to increase efficiency or the field's yield, in economic or environmental terms.

Until recently [13], farmers have mostly relied on their long-term experience on the particular acres. With the mentioned technology advances, sensors have cheapened data acquisition on such a scale that it makes them interesting for the data mining community. For carrying out an information-based field cultivation, the data have to be transformed into utilizable information in terms of management recommendations as a first step. This can be done by decision rules, which incorporate the knowledge about the coherence between sensor data and yield potential. In addition, these rules should give (economically) optimized recommendations. Since the data consist of simple and often even complete records of sensor measurements, there are numerous approaches known from data mining that can be used to deal with these data. One of those approaches are artificial neural networks [4] that may be used to build a model of the available data and help to extract the existing pattern. They have been used before in this context, e.g. in [1], [7] or [12].

The connection between information technology and agriculture is and will become an even more interesting area of research in the near future. In this context, IT mostly covers the following three aspects: data collection, analysis and recommendation [6]. This work is based on a dissertation that deals with data mining and knowledge discovery in precision agriculture from an agrarian point of view [15]. Hence this paper will also give a short overview of the previous work. On the other hand, since we are dealing with the above-mentioned data records, the computer science perspective will be applied. The main research target is whether we can model and optimize the site-specific data by means of computational intelligence techniques. We will therefore deal with data collection and analysis.

The paper is structured as follows: Section 2 will provide the reader with details on the acquisition of the data and some of the data's properties. Section 4 will give some background information on neural networks. In Section 5 we will describe the experimental layout and afterwards, we will evaluate the results that were obtained. The last section will give a brief conclusion.

2 Data Acquisition

The data available in this work have been obtained in the years 2003 and 2004 on a field near Köthen, north of Halle, Germany. All information available for this 65-hectare field was interpolated to a grid with 10 by 10 meters grid cell sizes. Each grid cell represents a record with all available information. During the growing season of 2004, the field was subdivided into different strips, where various fertilization strategies were carried out. For an example

of various managing strategies, see e.g. [11], which also shows the economic potential of PA technologies quite clearly. The field grew winter wheat, where nitrogen fertilizer was distributed over three application times.

Overall, there are seven input attributes – accompanied by the yield in 2004 as the target attribute. Those attributes will be described in the following. In total, there are 5241 records, thereof none with missing values and none with outliers.

2.1 Nitrogen Fertilizer – N1, N2, N3

The amount of fertilizer applied to each subfield can be easily measured. It is applied at three points in time into the vegetation period. Since the site of application had also been designed as an experiment for data collection, the range of N1, N2, and N3 in the data is from 0 to 100 $\frac{kg}{ha}$, where it is normally at around $60\frac{kg}{ha}$.

2.2 Vegetation – REIP32, REIP49

The *red edge inflection point* (REIP) is a first derivative value calculated along the red edge region of the spectrum, which is situated from 680 to 750nm. Dedicated REIP sensors are used in-season to measure the plants' reflection in this spectral band. Since the plants' chlorophyll content is assumed to highly correlate with the nitrogen availability (see, e.g. [10]), the REIP value allows for deducing the plants' state of nutrition and thus, the previous crop growth. For further information on certain types of sensors and a more detailed introduction, see [15] or [8]. Plants that have less chlorophyll will show a lower REIP value as the red edge moves toward the blue part of the spectrum. On the other hand, plants with more chlorophyll will have higher REIP values as the red edge moves toward the higher wavelengths. For the range of REIP values encountered in the available data, see Table 1. The numbers in the REIP32 and REIP49 names refer to the growing stage of winter wheat.

2.3 Electric Conductivity – EM38

A non-invasive method to discover and map a field's heterogeneity is to measure the soil's conductivity. Commercial sensors such as the EM-38[1] are de-

[1] trademark of Geonics Ltd, Ontario, Canada

signed for agricultural use and can measure small-scale conductivity to a depth of about 1.5 metres. There is no possibility of interpreting these sensor data directly in terms of its meaningfulness as yield-influencing factor. But in connection with other site-specific data, as explained in the rest of this section, there could be coherences. For the range of EM values encountered in the available data, see Table 1.

2.4 Yield 2003/2004

Here, wheat yield is measured in $\frac{t}{ha}$. In 2003, the range was from 1.19 to 12.38. In 2004, the range was from 6.42 to 11.37, with a higher mean and smaller standard deviation, see Table 1.

2.5 Data Overview

A brief summary of the available data attributes is given in Table 1.

Table 1 Data overview

Attribute	*min*	*max*	*mean*	*std*	*Description*
N1	0	100	57.7	13.5	amount of nitrogen fertilizer applied at the first date
N2	0	100	39.9	16.4	amount of nitrogen fertilizer applied at the second date
N3	0	100	38.5	15.3	amount of nitrogen fertilizer applied at the third date
REIP32	721.1	727.2	725.7	0.64	red edge inflection point vegetation index
REIP49	722.4	729.6	728.1	0.65	red edge inflection point vegetation index
EM38	17.97	86.45	33.82	5.27	electrical conductivity of soil
Yield03	1.19	12.38	6.27	1.48	yield in 2003
Yield04	6.42	11.37	9.14	0.73	yield in 2004

3 Points of Interest

From the agricultural perspective, it is interesting to see how much the influencable factor "fertilization" really determines the yield in the current site-year. Furthermore, there may be additional factors that correlate directly or indirectly with yield and which can not be discovered using regression or correlation analysis techniques like PCA. To determine those factors we could establish a model of the data and try to isolate the impact of single factors.

That is, once the current year's yield data can be predicted sufficiently well, we can evaluate single factors' impact on the yield.

From the data mining perspective, there are three points in time of fertilization, each with different available data on the field. What is to be expected is that, as more data is available, after each fertilization step the prediction of the current year's yield (`Yield04`) should be more precise. Since the data have been described in-depth in the preceding sections, Table 2 serves as a short overview on the three different data sets for the specific fertilization times.

Table 2 Overview on available data sets for the three fertilization times (FT)

Fertilization Time	*Available Sensor Data*
FT1	Yield03, EM38, N1
FT2	Yield03, EM38, N1, REIP32, N2
FT3	Yield03, EM38, N1, REIP32, N2, REIP49, N3

In each data set, the `Yield04` attribute is the target variable that is to be predicted. Once the prediction works sufficiently well and is reliable, the generation of, e.g., fertilization guidelines can be tackled. Therefore, the following section deals with an appropriate technique to model the data and ensure prediction quality.

4 Data Modeling

In the past, numerous techniques from the computational intelligence world have been tried on data from agriculture. Among those, neural networks have been quite effective in modeling yield of different crops ([12], [1]). In [14] and [15], artificial neural networks (ANNs) have been trained to predict wheat yield from fertilizer and additional sensor input. However, from a computer scientist's perspective, the presented work omits details about the ANN's internal settings, such as network topology and learning rates. In the following, an experimental layout will be given that aims to determine the optimal parameters for the ANN.

4.1 Neural Networks Basics

The network type which will be optimized here are multi-layer perceptrons (MLPs) with backpropagation learning. They are generally seen as a practical vehicle for performing a non-linear input-output mapping [4]. To counter the

issue of overfitting, which leads to perfect performance on training data but poor performance on test or real data, cross-validation will be applied. As mentioned in e.g. [5], the data will be split randomly into a training set, a validation set and a test set. Essentially, the network will be trained on the training set with the specified parameters. Due to the backpropagation algorithm's properties, the error on the training set declines steadily during the training process. However, to maximize generalization capabilities of the network, the training should be stopped once the error on the validation set rises [2].

As explained in e.g. [3], advanced techniques like Bayesian regularization [9] may be used to optimize the network further. However, even with those advanced optimization techniques, it may be necessary to train the network starting from different initial conditions to ensure robust network performance. For a more detailed and formal description of neural networks, we refer to [3] or [4].

4.2 Variable Parameters

For each network there is a large variety of parameters that can be set. However, one of the most important parameters is the network topology. For the data set described in Section 2, the MLP structure should certainly have up to seven input neurons and one output neuron for the predicted wheat yield. Since we are dealing with more than 5000 records, the network will require a certain amount of network connections to be able to learn the input-output mapping sufficiently well. Furthermore, it is generally unclear how many layers and how many neurons in each layer should be used [2]. Therefore, this experiment will try to determine those network parameters empirically. Henceforth, it is assumed that two layers are sufficient to approximate the data set. A maximum size of 32 neurons in the first and second hidden layer has been chosen – this provides up to 1024 connections in between the hidden layers, which should be sufficient. The range of the network layers' sizes will be varied systematically from 2 to 32. The lower bound of two neurons has been chosen since one neuron with a sigmoidal transfer function does not contribute much to the function approximation capabilities. Moreover, the network size has also been chosen for reasons of computation time.

4.3 Fixed Parameters

In preliminary experiments which varied further network parameters systematically, a learning rate of 0.5 and a minimum gradient of 0.001 have been found to deliver good approximation results without overfitting the data. All

of the network's neurons have been set to use the *tanh* transfer function, the initial network weights have been chosen randomly from an interval of $[-1, 1]$. Data have been normalized to an interval of $[0, 1]$.

4.4 Network Performance

The network performance with the different parameters will be determined by the mean of the squared errors on the test set since those test data will not be used for training. Overall, there are three data sets for which a network will be trained. The network topology is varied from 2 to 32 neurons per layer, leaving 961 networks to be trained and evaluated. The network's approximation quality can then be shown on a surface plot.

5 Results and Discussion

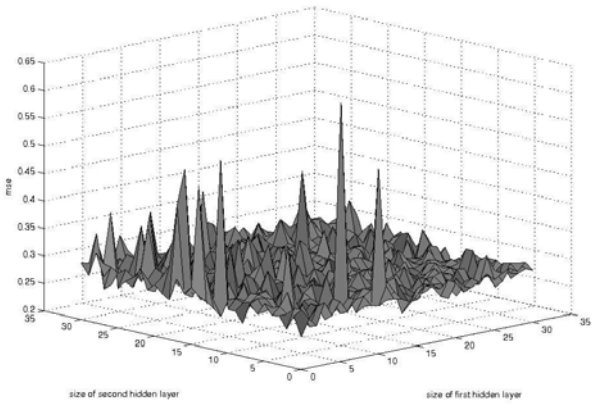


Fig. 1 MSE for first data set

To visualize the network performance appropriately, a surface plot has been chosen. In each of the following figures, the x- and y-axes show the sizes of the first and second hidden layer, respectively. Figures 1 and 2 show the mean squared error vs. the different network sizes, for two of the three fertilization times (FT), respectively. For the first FT, the mse on average is around 0.3, at the second FT around 0.25 and at the third FT around 0.2. It had been expected that the networks' prediction improves once more data (in

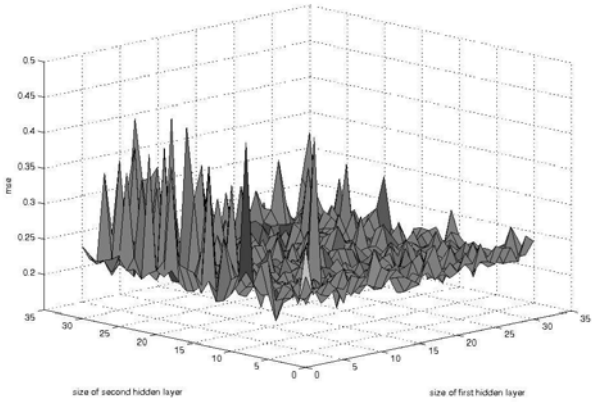


Fig. 2 MSE for third data set

terms of attributes) become available for training. There is, however, no clear tendency towards better prediction with larger network sizes. Nevertheless, a prediction accuracy of between 0.44 and 0.55 $\frac{t}{ha}$ (the figures only show the mean *squared* error) at an average yield of 9.14 $\frac{t}{ha}$ is a good basis for further developments with those data and the trained networks.

Furthermore, there are numerous networks with bad prediction capabilities in the region where the first hidden layer has much fewer neurons than the second hidden layer. Since we are using feedforward-backpropagation networks without feedback, this behaviour should also be as expected: the information that leaves the input layer is highly condensed in the first hidden layer if it has from two to five neurons – therefore, information is lost. The second hidden layer's size is then unable to contribute much to the network's generalization capabilities – the network error rises.

For the choice of network topology, there is no general answer to be given using any of the data sets from the different FTs. What can be seen is that the error surface is quite flat so that a layout with 16 neurons in both hidden layers should be an acceptable tradeoff between mean squared error and computational complexity. Furthermore, this choice is also substantiated by the fact that the variance of the mse during the cross-validation declines with larger hidden layer sizes.

Figure 3 shows the difference between the networks' mean squared errors vs. the different network sizes, respectively. For reasons of simplicity, the similar-looking plots for the differences between the networks trained on the first/second datasets as well as on the second/third dataset are not shown here. Figure 3 illustrates the networks' performance quite clearly. In the majority of cases, the networks generated from later data sets, i.e. those with

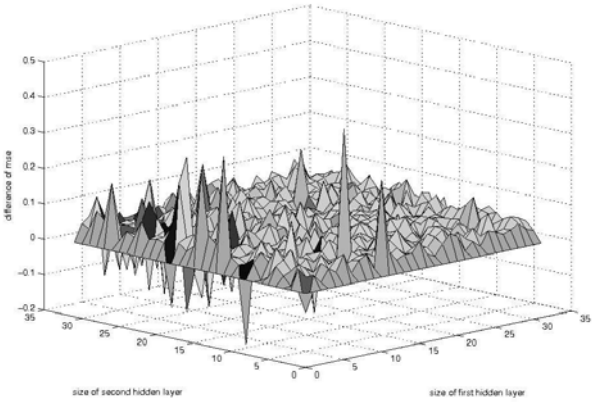


Fig. 3 MSE difference from first to third data set

more information, seem to be able to predict the target variable better than the networks from the earlier data sets.

6 Conclusion

This paper contributes to finding and evaluating models of agricultural yield data. Starting from a detailed data description, we built three data sets that could be used for training. In earlier work, neural networks had been used to model the data. Certain parameters of the ANNs have been evaluated, most important of which is the network topology itself. We built and evaluated different networks.

6.1 Future Work

In subsequent work, we will make use of the ANNs to model site-year data from different years. It will be evaluated whether the data from one year are sufficient to predict subsequent years' yields. It will also be interesting to study to which extent one field's results can be carried over to modeling a different field. The impact of different parameters during cropping and fertilization on the yield will be evaluated. Finally, controllable parameters such as fertilizer input can be optimized, environmentally or economically.

6.2 Acknowledgements

Experiments have been conducted using Matlab 2007b and the corresponding Neural Network Toolbox 5.1. The respective Matlab scripts to run the trials and generate the plots are available from the first author on request. The field trial data came from the experimental farm Görzig of Martin-Luther-University Halle-Wittenberg, Germany.

References

1. Drummond, S., Joshi, A., Sudduth, K.A.: Application of neural networks: precision farming. In: Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on, vol. 1, pp. 211–215 (1998)
2. Fausett, L.V.: Fundamentals of Neural Networks. Prentice Hall (1994)
3. Hagan, M.T.: Neural Network Design (Electrical Engineering). THOMSON LEARNING (1995)
4. Haykin, S.: Neural Networks: A Comprehensive Foundation (2nd Edition). Prentice Hall (1998)
5. Hecht-Nielsen, R.: Neurocomputing. Addison-Wesley (1990)
6. Heimlich, R.: Precision agriculture: information technology for improved resource use. Agricultural Outlook pp. 19–23 (1998)
7. Kitchen, N.R., Drummond, S.T., Lund, E.D., Sudduth, K.A., Buchleiter, G.W.: Soil Electrical Conductivity and Topography Related to Yield for Three Contrasting Soil-Crop Systems. Agron J **95**(3), 483–495 (2003)
8. Liu, J., Miller, J.R., Haboudane, D., Pattey, E.: Exploring the relationship between red edge parameters and crop variables for precision agriculture. In: Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International, vol. 2, pp. 1276–1279 vol.2 (2004)
9. MacKay, D.J.C.: Bayesian interpolation. Neural Computation **4**(3), 415–447 (1992)
10. Middleton, E.M., Campbell, P.K.E., Mcmurtrey, J.E., Corp, L.A., Butcher, L.M., Chappelle, E.W.: "Red edge" optical properties of corn leaves from different nitrogen regimes. In: Geoscience and Remote Sensing Symposium, 2002. IGARSS '02. 2002 IEEE International, vol. 4, pp. 2208–2210 vol.4 (2002)
11. Schneider, M., Wagner, P.: Prerequisites for the adoption of new technologies - the example of precision agriculture. In: Agricultural Engineering for a Better World. VDI Verlag GmbH, Düsseldorf (2006)
12. Serele, C.Z., Gwyn, Q.H.J., Boisvert, J.B., Pattey, E., Mclaughlin, N., Daoust, G.: Corn yield prediction with artificial neural network trained using airborne remote sensing and topographic data. In: Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000. IEEE 2000 International, vol. 1, pp. 384–386 vol.1 (2000)
13. Sonka, S.T., Bauer, M.E., Cherry, E.T., John, Heimlich, R.E.: Precision Agriculture in the 21st Century: Geospatial and Information Technologies in Crop Management. National Academy Press, Washington, D.C. (1997)
14. Wagner, P., Schneider, M.: Economic benefits of neural network-generated site-specific decision rules for nitrogen fertilization. In: J.V. Stafford (ed.) Proceedings of the 6th European Conference on Precision Agriculture, pp. 775–782 (2007)
15. Weigert, G.: Data Mining und Wissensentdeckung im Precision Farming - Entwicklung von ökonomisch optimierten Entscheidungsregeln zur kleinräumigen Stickstoffausbringung. Ph.D. thesis, TU München (2006)

Enhancing RBF-DDA Algorithm's Robustness: Neural Networks Applied to Prediction of Fault-Prone Software Modules

Miguel E. R. Bezerra[1], Adriano L. I. Oliveira[2], Paulo J. L. Adeodato[1], and Silvio R. L. Meira[1]

[1] Center of Informatics, Federal University of Pernambuco, P.O. Box 7851, 50.732-970, Cidade Universitaria, Recife-PE, Brazil {merb,pjla,srlm}@cin.ufpe.br
[2] Department of Computing Systems, Polytechnic School, University of Pernambuco, Rua Benfica, 455, Madalena, 50.750-410, Recife-PE, Brazil adriano@dsc.upe.br

Many researchers and organizations are interested in creating a mechanism capable of automatically predicting software defects. In the last years, machine learning techniques have been used in several researches with this goal. Many recent researches use data originated from NASA (National Aeronautics and Space Administration) IV&V (Independent Verification & Validation) Facility Metrics Data Program (MDP). We have recently applied a constructive neural network (RBF-DDA) for this task, yet MLP neural networks were not investigated using these data. We have observed that these data sets contain inconsistent patterns, that is, patterns with the same input vector belonging to different classes. This paper has two main objectives, (i) to propose a modified version of RBF-DDA, named RBF-eDDA (RBF trained with enhanced Dynamic Decay Adjustment algorithm), which tackles inconsistent patterns, and (ii) to compare RBF-eDDA and MLP neural networks in software defects prediction. The simulations reported in this paper show that RBF-eDDA is able to correctly handle inconsistent patterns and that it obtains results comparable to those of MLP in the NASA data sets.

1 Introduction

Machine learning techniques have already been used to solve a number of software engineering problems, such as software effort estimation [5], organization of libraries of components [17] and detection of defects in software [3, 9]. This paper is concerned with the detection of defects in software. We aim to predict if a software module contains some defect, without regard of how many defects it contains. For detecting defects in current software projects, a classifier needs to be trained previously with information about defects of past projects. In many papers, the data used during the experiments are obtained from a public repository made available by NASA [1] and by the Promise Repository [15].

Please use the following format when citing this chapter:

Bezerra, M.E.R., Oliveira, A.L.I., Adeodato, P.J.L. and Meira, S.R.L., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 119428.

These repositories contain static code measures and defect information about several projects developed by NASA.

Bezerra et. al [3] have evaluated the performance of some classifiers in the problem of defect detection. They reported that some patterns of the NASA data sets of the Promise Repository were inconsistent, that is, they had the same input vector and a class different from that of the other replicas. Inconsistent patterns are an inherent feature of some defect detection data sets and therefore should be handled appropriately by the classifiers. For instance, we can have two different software modules characterized by exactly the same features (such as (i) number of operators, (ii) number of operands, (iii) lines of code, etc). The problem is that the first can have defect and the other can be free of defects. This leads to inconsistent patterns (same input vector but different classes). Note that this situation arises because the input information regarding the software modules is not sufficient to differentiate them.

Bezerra et. al [3] have used the RBF-DDA classifier in their experiments and reported an important drawback in the DDA (Dynamic Decay Adjustment) algorithm: it does not work with inconsistent patterns. Therefore, in their experiments, patterns that had repetitions with conflicting classes (that is, inconsistent patterns) were discarded [3], which means loosing information. One of the contributions of this paper is to propose a modified version of RBF-DDA, referred to as RBF-eDDA, which is able to handle inconsistent patterns.

The DDA algorithm was originally proposed for constructive training of RBF neural networks [2]. The algorithm has achieved performance comparable to MLPs in a number of classification tasks and has a number of advantages for practical applications [2, 13], including the fact that it is a constructive algorithm which is able to build a network in only 4 to 5 epochs of training [2].

The use of neural networks for software defect prediction is rare in comparison to other techniques such as Decision Trees J4.8 [4, 7, 10, 12], k-Nearest Neighbor (kNN) [4, 7, 10], and Naive Bayes [7, 10, 12]. Furthermore, multi-layer perceptron (MLP) neural networks was not used for the detection of fault-prone modules using the NASA data. MLPs were used for software defect prediction, yet using other data (not from NASA), such as in [9]. In this way, our experiments utilize MLP neural networks trained with backpropagation to evaluate its performance in the detection of software defects and to compare it to RBF-eDDA in this task.

In summary, the main contributions of this paper are: (i) to propose a modified version of the RBF-DDA algorithm, named RBF-eDDA, which aims to handle inconsistent patterns, (ii) to apply RBF-eDDA to software defect prediction in the NASA data sets, and (iii) to apply MLP to software defect prediction in the NASA data sets and to compare the results obtained to those of RBF-eDDA.

The rest of this paper is organized as follows. Section 2 briefly reviews the standard RBF-DDA network and describes the proposed method, RBF-eDDA. Sections 3 presents the methods used to assess and compare the classifiers whereas Section 4 presents the experiments and discusses the results obtained.

Finally, the Section 5 presents the conclusions and suggestions for future research.

2 The Proposed Method

RBF-DDA neural networks have a single hidden layer, whose number of units is automatically determined during training. In this way, during training, the topology starts with an empty hidden layer. Next, the neurons are dynamically included on it until a satisfactory solution has been found [2, 13]. The activation $R_i(\overrightarrow{x})$ of a hidden neuron i is given by the Gaussian function (Eq. 1), where $\overrightarrow{x}$ is the input vector, $\overrightarrow{r_i}$ is the center of the ith Gaussian and σ_i denotes its standard deviation, which determines the Gaussian's width.

$$R_i(\overrightarrow{x}) = exp\left(-\frac{\|\overrightarrow{x} - \overrightarrow{r_i}\|^2}{\sigma_i^2}\right) \tag{1}$$

RBF-DDA uses 1-of-n coding in the output layer, with each unit of this layer representing a class. Classification uses a *winner-takes-all* approach. In this way, the output unit with the highest activation gives the class. Each hidden unit is connected to exactly one output unit and has a weight A_i, whose value is determined by the training algorithm. Output units use linear activation functions with values computed by $f(\overrightarrow{x}) = \sum_{i=1}^{m} A_i \times R_i(\overrightarrow{x})$, where m is the number of RBFs connected to that output. In this paper, each output is normalized as proposed by et al. Bezerra in [3]. Thus, the RBF-DDA becomes capable to produce continuous outputs that represent the probability of a module being fault-prone.

The DDA algorithm has two parameters, namely, θ^+ and θ^-, whose default values are 0.4 and 0.1, respectively [2]. These parameters are used to decide on the introduction of new neurons in the hidden layer during training. The DDA training algorithm for one epoch is presented in the Algorithm 1 [2].

It was originally believed that the parameters θ^+ and θ^- would not influence RBF-DDA performance. Yet, Oliveira et al. have recently demonstrated that the value of θ^- may significantly influence classification performance [13]. Therefore, in this paper we use the default value for θ^+ ($\theta^+ = 0.4$) and select the best value of θ^- for each data set via cross-validation, as in [13].

2.1 Training RBF Networks with the Enhanced DDA Algorithm

Before the explanation of the modifications in the DDA algorithm, it is necessary to understand its drawbacks regarding inconsistent patterns. For this task, we use an example that has a one-dimensional training set composed by

Algorithm 1 DDA algorithm (one epoch) for RBF Training

1: **for all** prototypes p_i^k **do** ▷ Reset weights
2: $A_i^k = 0.0$
3: **end for**
4: **for all** training pattern $(\overrightarrow{x}, c)$ **do** ▷ Train one complete epoch
5: **if** $\exists p_i^c : R_i^c(\overrightarrow{x}) \geq \theta^+$ **then**
6: $A_i^c += 1.0$
7: **else**
8: add new prototype $p_{m_c+1}^c$ with: ▷ Commit: introduce new prototype
9: $\overrightarrow{r}^c_{m_c+1} = \overrightarrow{x}$
10: $\sigma^c_{m_c+1} = max_{k \neq c \wedge 1 \leq j \leq m_k}\{\sigma : R^c_{m_c+1}(\overrightarrow{r}^k_j) < \theta^-\}$
11: $A^c_{m_c+1} = 1.0$
12: $m_c += 1$
13: **end if**
14: **for all** $k \neq c, 1 \leq j \leq m_k$ **do** ▷ Shrink: adjust conflicting prototypes
15: $\sigma_j^k = max\{\sigma : R_j^k(\overrightarrow{x}) < \theta^-\}$
16: **end for**
17: **end for**

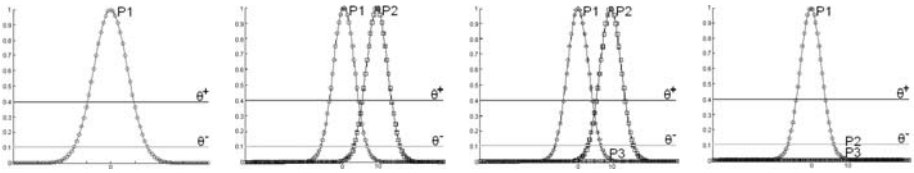
three patterns: (*P1*)=0, class=$C1$; (*P2*)=10, class=$C0$; (*P3*)=10, class=$C1$. The patterns *P2* and *P3* have the same input, but are from distinct classes.

Following Algorithm 1, when *P1* is presented, there is no neuron in the hidden layer, and then a new Gaussian (RBF) is created with $\overrightarrow{r_{P_1}} = [0]$, $A_{P_1} = 1$ and $class = C1$ (Fig. 1(a)). When *P2* is encountered, the DDA algorithm introduces a new Gaussian with $\overrightarrow{r_{P_2}} = [10]$, $A_{P_2} = 1$ and $class = C0$, since there was no neuron from class $C0$.

After *P2*'s Gaussian have been included, all others Gaussians with class different from $C0$ have their width shrank (Fig. 1(b)). When *P3* is presented, the algorithm realizes that it has the same class of *P1* and that its activation is less than θ^+, then a new prototype should be introduced. At this moment, DDA's drawback can be observed. The *P3*'s Gaussian conflicts with that of *P2*, because both have the same center but distinct classes. Thus, the Euclidean distance between the centers of *P2* and *P3* is *zero*. In other words, Eq. 1 gives $\|\overrightarrow{x_{P_3}} - \overrightarrow{r_{P_2}}\| = 0$, and that makes the standard deviation of *P3* equal to *zero* as well ($\sigma_{P_3} = 0$), causing a division by *zero* (see line 10 of Algorithm 1). The result can be seen in Fig. 1(c).

After *P3* is introduced, the algorithm shrank the width of Gaussians with a class different of $C1$. In this way, σ_{P_2} receive *zero* because $\|\overrightarrow{x_{P_3}} - \overrightarrow{r_{P_2}}\| = 0$, then it will happen to *P2* the same as *P3*, as can be seen in Fig. 1(d). Therefore, the algorithm never converges to one solution, because the final SSE (Sum of Squared Errors) is always calculated as *NaN* (Not-a-Number).

To handle the problem caused by inconsistent patterns just described, we propose a modified version of RBF-DDA, named RBF-eDDA, which aims to turn it robust enough to treat the problem of inconsistent patterns. We began from the principle that the hidden layer could not hold this type of conflict, because it will influence the inclusion of new Gaussians and the adjustment of its



(a) (b) (c) (d)

Fig. 1 An example of the DDA Algorithm's drawback: (a) *P1* is encountered and a new RBF is created; (b) *P2* leads to a new prototype for class $C0$ and shrinks the radius of the existing RBFs of class $C1$; (c) *P3* Gaussian is presented and cannot be calculated because $\sigma_{P_3} = 0$; (d) during the shrink of the Gaussians, $\sigma_{P_2} = 0$ and *P2* also cannot be calculated.

width. In RBF-eDDA, whenever a new Gaussian is included, the algorithm will verify if it is conflicting with some pre-existent Gaussian. If it is not conflicting, the algorithm will create the Gaussian normally; otherwise, the algorithm will not create a new Gaussian. Instead, the algorithm will reduce the weight $A_i = A_i - 1$ of the pre-existent Gaussian that is conflicting. If the Gaussian weight becomes *zero*, it is removed from the hidden layer.

3 Assessing the Performance

The defect detection is a cost-sensitive task whereby a misclassification is more costly than correct classification. Other problem is that the data sets utilized to train the predictors have a skewed class distribution, that is, these data sets have more modules with defects than modules without defects. Then, we need to use evaluation metrics capable to assess the performance of the classifiers, and that handle these constraints. The ROC curve [18] is the best way to deal with cost-sensitive problems and unbalanced datasets because it depicts the performance of a classifier regardless of the class distribution or the error costs [18]. Thus, in order to assess the classifiers, we use ROC's AUC (Area Under Curve). The best classifier is the one with the higher AUC [18].

A defect predictor is a binary classifier that has four possible outcomes, as shown in Fig. 2, which depicts the confusion matrix. Considering the defect detection problem, the columns of the Fig. 2 represent the actual class of a software module while the rows represent the class predicted by the classifier. Thus, the *NO column* represents the modules that do not have defects while the *YES column* represents the inverse. Conversely, the *no row* represents the modules labeled as fault-free by the classifier, while the *yes row* represents the modules labeled as fault-prone.

The confusion matrix is the core of several evaluation metrics. The *Accuracy* (Acc) is the proportion of the total number of modules that were correctly classified (Eq. 2). The *Probability of Detection* (PD) is the proportion of defective

		Actual Class	
		NO	**YES**
Prediced Class	**no**	True Negative	False Negative
	yes	False Positive	True Positive

Fig. 2 Confusion Matrix of a Binary Classifier.

modules that were correctly identified (Eq. 3). In the other case, the *Probability of False Alarm* (PF) (Eq. 4), is the proportion of correct modules that were incorrectly identified. Another metric is the *Precision* (Prec), that is the proportion of the predicted defective modules that were correct (Eq. 5).

$$Acc = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{2}$$

$$PD = \frac{TP}{(TP + FN)} \tag{3}$$

$$PF = \frac{FP}{(FP + TN)} \tag{4}$$

$$Prec = \frac{TP}{(TP + FP)} \tag{5}$$

These four metrics are used during our experiments to evaluate the performance, yet their values depend on the adjustment of the classifiers' operation point (threshold), because it modifies the class memberships and, consequently, the confusion matrix distribution. To choose the classifier's best threshold, we also use the ROC curves; the threshold is the best ROC's point, which is the one closer to the point (x-*axis*=0, y-*axis*=1) [6].

4 Experiments

This Section presents the experiments carried out with the proposed method, RBF-eDDA, as well as with MLP neural networks. In the experiments with RBF-eDDA, we adjusted the parameter θ^- to find the best performance of the classifier [13]. We employed the following values for this parameter: 0.2, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5} and 10^{-6}. These values were chosen because they were used successfully in previous papers [13, 3].

In this study, we compare the performance of RBF-eDDA with other neural network, the MLP trained with Backpropagation [8]. MLP is a feedforward neural network. As its architecture is not constructive, it is necessary to vary its topology to choose the one that performs well for a given problem. In the experiments we varied three parameters of the MLP: the number of neurons

in the hidden layer, the *learning rate* (η) and the *momentum*. The simulations using MLP networks were carried out using different topologies, that is, number of neurons in the hidden layer. The values used were: 1, 3, 5, 9 and 15. The values used for the *learning rate* were 0.001 and 0.01; for the *momentum* we used 0.01 and 0.1.

In our experiments we are concerned with the reproducibility, then, we decided that the five data sets used – $CM1$, $JM1$, $KC1$, $KC2$ and $PC1$ – should be obtained from the Promise Repository [15], since it stores data sets that can be reused by other researches. Each dataset has 21 input features based on software metrics such as cyclomatic complexity, lines of comments, total operators, total lines of codes, etc. Detailed information about each feature can be obtained freely on the MDP web site [1].

To compare two or more classifiers, it is important that the data used in the evaluation are exactly the same. Then, to guarantee the reproducibility of our results, all experiments utilized the same data set separation. In order to do this, the fold separation and the stratification were made by the Weka [18]. We set the Weka's random seed to 1 and made the separation of the datasets in 10 stratified folds. Then, using the Weka framework and Promise datasets, other researchers can reproduce the experiments reported here.

Before the simulations, we made a preprocessing in the datasets. This is a procedure whereby the data are cleaned and prepared for training and for testing the classifiers. Initially we observed that some patterns had missing values. These patterns with missing values were removed, since they represented a very small sample of the total number of patterns in each data set (less than 0.1%) and therefore could be discarded [11]. The next step of the preprocessing was the dataset normalization because the values of each feature had different amplitudes in relation to the others, and this could induce skewed results. Table 1 summarizes the characteristics of each dataset after preprocessing. Notice that all datasets have a small percentage of modules with some defect.

Table 1 Summary of the datasets used in this paper.

Dataset	#Modules	%Modules with defects
CM1	498	9.83
JM1	10885	19.35
KC1	2109	15.45
KC2	522	20.50
PC1	1109	6.94

4.1 Analysis of Results

In our comparison between RBF-eDDA and MLP, the AUC is used as the main criterion; we also report the PD, PF, Acc and Prec obtained by the classifiers for comparison. The AUC was selected as the most important criterion because it summarizes the global performance of the classifier in a single scalar value. The simulations' results of the RBF-eDDA and MLP are reported in Table 2.

Table 2 Results of the classifiers RBF-eDDA and MLP with backpropagation.

	RBF-eDDA							MLP with BackPropagation				
Dataset	θ^-	#Units	AUC	PD	PF	Acc	Prec	AUC	PD	PF	Acc	Prec
CM1	10^{-3}	336	0.773	0.837	0.347	0.671	0.208	0.760	0.755	0.327	0.681	0.201
JM1	10^{-2}	7238	0.596	0.653	0.461	0.561	0.254	0.718	0.652	0.326	0.670	0.324
KC1	10^{-5}	999	0.712	0.801	0.384	0.644	0.276	0.792	0.755	0.309	0.701	0.309
KC2	10^{-1}	252	0.744	0.766	0.313	0.703	0.387	0.825	0.757	0.186	0.803	0.513
PC1	10^{-2}	613	0.859	0.805	0.218	0.784	0.216	0.819	0.805	0.300	0.707	0.167
Average			0.737	0.772	0.345	0.673	0.268	0.783	0.745	0.289	0.712	0.303

For RBF-eDDA, Table 2 shows the best θ^- for each dataset and the number of neurons of the hidden layer. For the MLP networks the configuration that obtained the best stability and performance across all datasets was the configuration with 3 neurons in the hidden layer, *learning rate* set to 0.01 and *momentum* set to 0.1. Using the AUC for the comparison, notice that the MLP outperformed the RBF-eDDA classifier on the JM1, KC1 and KC2 datasets; on the other hand, the RBF-eDDA outperformed the MLP in the CM1 and the PC1. In the CM1 dataset, the difference between the classifiers was small, but in the PC1 dataset it was higher.

Fig. 3 shows the ROC curve of the classifiers for each dataset along with the best operation points. The values of PD, PF, Acc and Prec were computed using these operation points. A prominent result of the MLP has occurred in the KC2 dataset, with PD=75.7% and a high accuracy and small amount false alarms. In the case of RBF-eDDA, the best result occurred in the PC1 dataset, with a PD=80.5%, PF=21.8% and a high accuracy.

5 Conclusions

This paper contributes by proposing RBF-eDDA, an enhanced version of the DDA algorithm. RBF-eDDA aims to handle inconsistent patterns, which occur in software defect detection data sets. The original RBF-DDA algorithm was not able to train if the data set contains inconsistent patterns. We report a number of experiments that have shown that RBF-eDDA handles inconsistent patterns

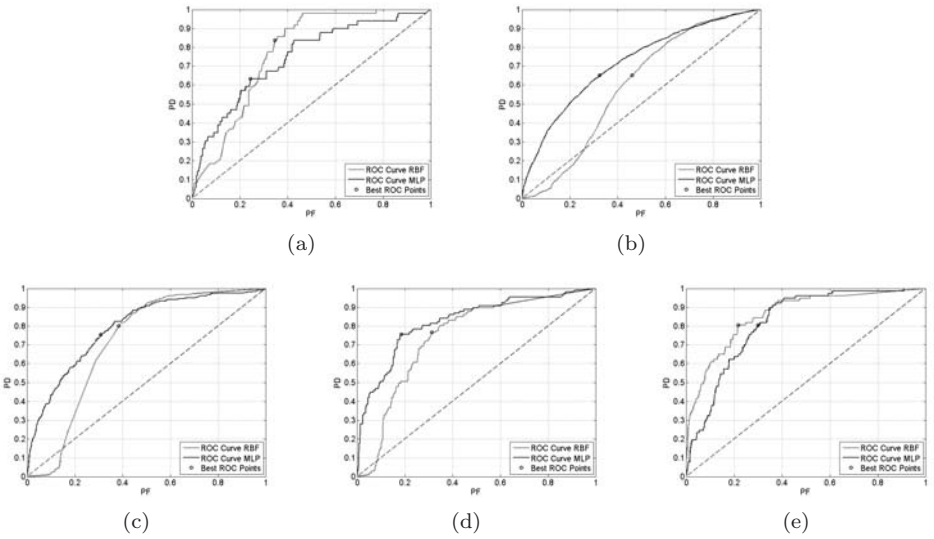


Fig. 3 ROC Curves obtained by the MLP and RBF-eDDA classifiers for the datasets CM1(a), JM1(b), KC1(c), KC2(d) and PC1(e).

adequately. Our experiments also aimed to compare the proposed method to MLP networks for software defect prediction. The experiments have shown that RBF-eDDA and MLP have similar performance in this problem. RBF-eDDA offers an advantage over MLP since it has only one critical parameter (θ^-) whereas MLP has three.

Considering the study of Shull et al. [16], which asserts that in a real development environment a peer review catches between 60-90% of the defects, our results are useful, since RBF-eDDA achieved mean PD of 77.2% and the MLP achieved 74.5%.

We endorse the conclusions of Menzies et al.[12], since they state that these predictors would be treat as *indicators* and not as definitive *oracles*. Therefore, the predictors are suitable tools to guide test activities, aiding on the prioritization of resources and, hence, in the reduction of costs in software factories where the development resources are scarce.

As future work, we propose to investigate a committee machine composed of RBF-eDDA and MLP networks to achieve a better classification performance; this was already used with success in time series novelty detection [14]. The motivation for such a committee is that in some data sets RBF-eDDA outperforms MLP whereas in others the inverse occurs.

Acknowledgments

The authors would like to thank the CNPq (Brazilian Research Agency) for its financial support. They also would like to thank NeuroTech (Brazilian Data Mining Company) for allowing the use of its MLP simulator.

References

1. Metrics data program. URL `http://mdp.ivv.nasa.gov`
2. Berthold, M.R., Diamond, J.: Boosting the performance of RBF networks with dynamic decay adjustment. In: Adv. in Neural Inf. Proc. Syst., vol. 7, pp. 521–528 (1995)
3. Bezerra, M.E.R., Oliveira, A.L.I., Meira, S.R.L.: A constructive RBF neural network for estimating the probability of defect in software modules. In: IEEE Int. Joint Conference on Neural Networks, pp. 2869–2874. Orlando, USA (2007)
4. Boetticher, G.D.: Nearest neighbor sampling for better defect prediction. SIGSOFT Softw. Eng. Notes **30**(4), 1–6 (2005)
5. Braga, P.L., Oliveira, A.L.I., Ribeiro, G., Meira, S.R.L.: Bagging predictors for estimation of software project effort. In: IEEE International Joint Conference on Neural Networks (IJCNN2007), pp. 1595–1600. Orlando-Florida, USA (2007)
6. Fawcett, T.: An introduction to ROC analysis. Pattern Recognition Letters **27**(8), 861–874 (2006)
7. Guo, L., et al.: Robust prediction of fault-proneness by random forests. In: 15th International Symposium on Software Reliability Engineering, pp. 417–428 (2004)
8. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice Hall (1998)
9. Kanmani, S., et al.: Object-oriented software fault prediction using neural networks. Inf. Softw. Technol. **49**(5), 483–492 (2007)
10. Khoshgoftaar, T.M., Seliya, N.: The necessity of assuring quality in software measurement data. In: IEEE METRICS, pp. 119–130. IEEE Computer Society (2004)
11. Lakshminarayan, K., Harp, S.A., Samad, T.: Imputation of missing data in industrial databases. Appl. Intell. **11**(3), 259–275 (1999)
12. Menzies, T., Greenwald, J., Frank, A.: Data mining static code attributes to learn defect predictors. IEEE Trans. Software Engineering **33**(1), 2–13 (2007)
13. Oliveira, A.L.I., et al.: On the influence of parameter θ^- on performance of RBF neural networks trained with the dynamic decay adjustment algorithm. Int. Journal of Neural Systems **16**(4), 271–282 (2006)
14. Oliveira, A.L.I., Neto, F.B.L., Meira, S.R.L.: Combining MLP and RBF neural networks for novelty detection in short time series. In: MICAI, Lecture Notes in Computer Science, vol. 2972, pp. 844–853. Springer (2004)
15. Shirabad, J., Menzies, T.: The promise repository of software engineering databases (2005). URL `http://promise.site.uottawa.ca/SERepository`
16. Shull, F., et al.: What we have learned about fighting defects. In: Eighth IEEE Int. Symposium on Software Metrics, pp. 249–258 (2002)
17. Veras, R.C., de Oliveira, A.L.I., de Moraes Melo, B.J., de Lemos Meira, S.R.: Comparative study of clustering techniques for the organization of software repositories. In: IEEE Int. Conf. on Tools with Artificial Intelligence, pp. 210–214 (2007)
18. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann (2005)

LEARNING

A Study with Class Imbalance and Random Sampling for a Decision Tree Learning System

Ronaldo C. Prati, Gustavo E. A. P. A. Batista, and Maria Carolina Monard

Abstract Sampling methods are a direct approach to tackle the problem of class imbalance. These methods sample a data set in order to alter the class distributions. Usually these methods are applied to obtain a more balanced distribution. An open-ended question about sampling methods is which distribution can provide the best results, if any. In this work we develop a broad empirical study aiming to provide more insights into this question. Our results suggest that altering the class distribution can improve the classification performance of classifiers considering AUC as a performance metric. Furthermore, as a general recommendation, random over-sampling to balance distribution is a good starting point in order to deal with class imbalance.

1 Introduction

A key point for the success of Machine Learning – ML – application in Data Mining is related to understanding and overcoming some practical issues that have not been previously considered when learning algorithms were initially proposed. One of these issues that has come into light in supervised learning is related to class imbalance, where some classes are represented by a large number of examples while the others are represented by only a few. Numerous studies report a poor performance of the induced models in domains where class imbalance is present [2, 10].

Sampling methods are a direct approach to tackle the problem of class imbalance. These methods sample a data set in order to alter the class distributions. Usually these methods are applied in order to obtain a more bal-

University of São Paulo
P. O. Box 668, ZIP Code 13560-970
São Carlos (SP), Brazil
e-mail: {prati,gbatista,mcmonard}@icmc.usp.br

Please use the following format when citing this chapter:

Prati, R.C., Batista, G.E.A.P.A. and Monard, M.C., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 131140.

anced distribution. The two most well-known sampling methods to deal with the problem of class imbalance are random over-sampling and random under-sampling. These methods replicate (eliminate) examples of the minority (majority) class in order to obtain a more balanced distribution.

An open-ended question considering sampling methods is which distribution can provide the best results, if any. In this work we develop a broad empirical study aiming to provide more insights into this question. To this end, random under-sampling and random over-sampling methods were used to change the class distribution of fourteen UCI [1] data sets. The data sets are under and over-sampled to reach thirteen different fixed class distributions and used as input to a decision tree learning algorithm (C4.5) to induce a model. Our results suggest that altering the class distribution can improve the classification performance of classifiers considering AUC as the performance metric.

Furthermore, as a general recommendation given the results obtained, random over-sampling can be considered a good starting point to deal with class imbalance. This method is straightforward to implement and considerably fast if compared with more sophisticated (heuristic) sampling methods. Over-sampling attempting to reach the balanced distribution is also a good first choice as AUC values near the balanced distribution are often the best.

This work is organized as follows: Section 2 presents some notes on ROC analysis, and its importance in evaluating the performance of classifiers in imbalanced domains. Section 3 discusses our methodology and the experimental results obtained. Finally, Section 4 presents some concluding remarks as well as outlines some future research.

2 ROC analysis

From hereafter, we constrain our analysis to two-class problems, where the minority class will be called **positive** and the majority class **negative**.

A straightforward connection between class imbalance and error rate might be traced by observing that it is easier to achieve a low overall error rate by simply predicting the majority class. For instance, it is straightforward to create a classifier having an error rate of 1% in a domain where the majority class proportion corresponds to 99% of the instances, by simply forecasting every new example as belonging to the majority class.

In scenarios where the target class priors and/or misclassification costs are unknown or are likely to change, the use of error rate as a basic performance measure may lead to misleading conclusions. This is due to the fact that the error rate strongly depends on class distribution and misclassification costs. Furthermore, the use of the error rate in such conditions does not allow the direct comparison/evaluation of how learning algorithms would perform in different scenarios. In a nutshell, two fundamental aspects of performance,

namely discrimination capacity and decision tendencies[1], are confused when error rate is used as a basic performance measure.

Often, we are primarily interested in the discrimination aspect. In this case we want to leave out the decision aspect such that it does not mislead the evaluation of classifiers. Receiver Operating Characteristic (ROC) analysis [8] provides such a way of assessing a classifier performance independently of the criterion adopted for making a particular decision on how to trade-off true/false positives as well as the bias used by learning algorithms toward one particular decision or another. Thus, ROC based methods provide a fundamental tool for analyzing and assessing classifiers performance in imprecise environments. The basic idea is to decouple relative error rate (percentage of false positives or false positive rate – FP_{rate}) from hit rate (percentage of true positives or true positive rate – TP_{rate}) by using each of them as axis in a bi-dimensional space. Thus, in ROC analysis a classifier is represented by a pair of values instead of a single error rate value. Furthermore, spreading the classifier criterion over all possible trades off of hits and errors, a curve that works as an index that reflects the subjective probabilities and utilities that determine all possible criteria is obtained.

For instance, considering a classifier that provides probabilities of an example belonging to each class, such as the Naive Bayes classifier, we can use these probabilities as a threshold parameter biasing the final class selection. Then, for each threshold, we plot the percentage of hits against the percentage of errors. The result is a bowed curve, rising from the lower left corner (0,0), where both percentages are zero, to the upper right corner (1,1), where both percentages are 100%. The more sharply the curve bends, the greater the ability of coping with different class proportions and misclassification costs, since the number of hits relative to the number of false alarms is higher. By doing so, it is possible to consider what might happen if a particular score is selected as a classification threshold, allowing to select the most suitable threshold given a specific situation.

In situations where neither the target cost distribution nor the class distribution are known, an alternative metric to compare models through ROC analysis is the area under the ROC curve (AUC). The AUC represents the probability that a randomly chosen positive example will be rated higher than a negative one [12], and in this sense it is equivalent to the Wilcoxon test of ranks. However, it should be kept in mind that given a specific target condition, the classifier with the maximum AUC may not be the classifier with the lowest error rate.

[1] Discrimination capacity can be defined as how well the system is able to discriminate between positive and negative examples. Decision tendencies can be understood as how well the system is able to manage the trade-off between true and false positives given different misclassification costs and class distribution scenarios.

3 Experiments

The experiments involved the application of two sampling methods to fifteen UCI [1] data sets. We start describing the sampling methods and the methodology used in the experiments, followed by an analysis of the results obtained.

The two sampling methods used in the experiments with the objective of altering the class distribution of training data are:

Random under-sampling: a method that reduces the number of examples of one of the classes through the random elimination of examples of this class.

Random over-sampling: a method that increases the number of examples of one of the classes through the random replication of examples of this class.

Usually, random under-sampling and random over-sampling are used to approximate the prior probabilities of each class. Therefore, random under-sampling is usually applied to the majority (negative) class while random over-sampling is usually applied to the minority (positive) class.

Several authors agree that the major drawback of random under-sampling is that this method can discard potentially useful data that could be important to the induction process. On the other hand, random over-sampling supposedly increases the likelihood of occurring overfitting, since it makes exact copies of the minority class examples. For instance, a symbolic classifier might construct rules that are apparently accurate although actually cover one replicated example.

For experimental analysis, we selected fourteen data sets from UCI [1] having different degrees of imbalance. Table 1 summarizes the data sets used in this study. For each data set, it shows the number of examples (#Examples), number of attributes (#Attributes), together with the number of quantitative and qualitative attributes in brackets, class labels and class distribution. For data sets having more than two classes, we chose the class with fewer examples as the positive class, and collapsed the remainder as the negative class.

Our implementation of random over-sampling and random under-sampling methods have a parameter that allows the user to set up the desired class distribution that should be reached after the application of these methods. We over and under-sampled all data sets until the following positive class distributions were reached: 5%, 7.5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 92.5% and 95%. Distributions greater than 50% mean that the application of the over or under-sampling methods made the positive class more frequent than the negative class. Moreover, in order to reach distributions smaller (more imbalanced) than the original ones, we over-sampled the negative class or under-sampled the positive class, depending on which method was being applied.

Table 1 Data sets summary descriptions.

Data set #	Name	#Examples	#Attributes (min., maj.)	Class (positive,negative)	Class proportion
1	sonar	208	61 (61,0)	(m, r)	(46.6%, 53.5%)
2	heart	270	14 (14,0)	(2, 1)	(44.4%, 55.6%)
3	bupa	345	7 (7,0)	(1, 2)	(42.0%, 58.0%)
4	ionosphere	351	34 (34,0)	(bad, good)	(35.9%, 64.1%)
5	breast	683	10 (10,0)	(malignant, benign)	(35.0%, 65.0%)
6	pima	768	8 (8,0)	(1, 0)	(34.8%, 65.2%)
7	tic-tac-toe	958	10 (0,10)	(positive, negative)	(34.7%, 65.3%)
8	german	1000	20 (7,13)	(bad, good)	(30.0%, 70.0%)
9	haberman	306	3 (3,0)	(die, survive)	(26.5%, 73.5%)
10	vehicle	846	18 (18,0)	(van, remainder)	(23.5%, 76.5%)
11	new-thyroid	215	5 (5,0)	(hypo, remainder)	(16.3%, 83.7%)
12	ecoli	336	7 (7,0)	(imu, remainder)	(10.4%, 89.6%)
13	flag	194	28 (10,18)	(white, remainder)	(8.8%, 91.2%)
14	glass	214	9 (9,0)	(ve-win-float-proc, remainder)	(7.9%, 92.1%)

It is important to note that, as under and over-sampling are applied, the number of training examples will vary. In particular, the number of training examples of under-sampled data sets might be significantly reduced. This shortcoming is one of the most frequent criticisms regarding under-sampling as this method might discard important information.

It should be observed that our experimental setup is significantly different from [14], where Weiss & Provost consider a scenario in which data are expensive to acquire, and they analyze the effect of class distribution. Their experimental setup uses random under-sampling, however training set sizes are constant for all class distributions.

In our experiments release 8 of the C4.5 [13] symbolic learning algorithm was used to induce decision trees. The trees were induced with default parameter settings. m-estimation [4] was used to improve the leaf probability estimates to produce ROC curves. We adjusted the m parameter so that $bm = 10$ as suggested in [6], where b is the prior probability of the positive class. We also use the AUC as the main method to assess our experiments.

Table 2 presents the AUC values obtained by the trees induced by C4.5 with random under and over-sampled data sets. The first column in Table 2 specifies the number of the data set (according to Table 1) and the next two columns specify the natural proportion of positive examples followed by the AUC values assessed using this distribution. The next columns present the AUC values for the thirteen fixed class distributions. Each line has been split into two, each one presenting the results obtained with random over and under-sampled data sets, as indicated in the fourth column. All AUC values in Table 2 were obtained using 10-fold stratified cross-validation, and the values between brackets refer to the standard deviations.

The highest AUC values for each data set/method are shaded. The last column (p-value) shows the p-value of the statistical test comparing the shaded results with the natural distribution. The statistical procedure used to carry out the tests is the Student paired t-test, with the null hypothsis (H_0) that

Table 2 AUC results for random under (Ud) and over (Ov) sampling methods for several positive class distributions.

#	Original		Smp	Proportion of positive examples													*p*-value
	Prp	AUC		95%	92.5%	90%	80%	70%	60%	50%	40%	30%	20%	10%	7.5%	5%	
1	46.6	79.0(11.0)	Ov	68.4(9.4)	68.5(10.0)	73.8(12.3)	77.5(11.3)	71.5(12.5)	80.6(10.8)	79.1(12.6)	77.7(7.3)	73.8(12.1)	70.9(13.1)	73.5(12.6)	71.5(8.4)	74.3(12.5)	0.37
			Ud	53.3(10.6)	55.0(12.3)	56.0(8.4)	75.7(17.3)	77.5(15.6)	72.7(17.9)	78.4(10.6)	78.8(14.9)	70.6(12.4)	65.9(9.9)	66.8(12.2)	65.5(11.7)	62.3(10.0)	0.49
2	44.4	85.5(10.4)	Ov	82.2(14.3)	84.5(14.3)	87.3(10.1)	88.1(10.3)	87.2(9.4)	84.1(10.0)	84.2(9.0)	85.6(11.0)	87.6(9.1)	84.7(10.3)	86.1(12.4)	87.6(9.4)	87.3(8.6)	0.29
			Ud	53.1(9.9)	62.5(16.3)	74.0(20.7)	79.5(11.9)	84.9(9.8)	86.8(8.8)	87.0(6.9)	88.4(10.6)	88.8(8.2)	82.6(11.4)	70.8(8.5)	65.7(10.0)	59.1(9.2)	0.22
3	42.0	65.2(6.6)	Ov	63.1(6.8)	65.6(6.0)	65.1(9.6)	65.4(8.6)	61.6(6.6)	64.1(9.5)	68.0(9.6)	64.5(7.4)	65.1(6.5)	66.3(4.6)	67.1(4.5)	65.0(8.3)	66.2(7.7)	0.23
			Ud	50.0(0.0)	49.6(1.5)	52.7(7.1)	57.1(7.2)	61.3(10.8)	61.8(9.1)	64.0(6.6)	62.9(6.2)	61.1(7.4)	55.4(8.6)	57.3(9.2)	53.2(7.9)	50.8(2.5)	0.22
4	35.9	91.4(5.0)	Ov	89.4(6.6)	89.7(5.8)	91.8(4.6)	92.2(4.9)	91.9(4.3)	92.7(4.7)	92.8(6.4)	92.7(3.7)	91.3(5.1)	92.9(4.1)	90.2(5.3)	90.9(5.8)	89.4(6.6)	0.24
			Ud	66.0(10.6)	75.8(5.9)	79.6(6.0)	88.2(6.2)	89.6(5.8)	92.0(5.2)	90.2(6.0)	90.4(4.4)	89.1(4.2)	89.2(5.2)	85.8(5.9)	83.6(6.7)	82.1(9.3)	0.40
5	35.0	95.7(3.9)	Ov	97.1(2.7)	97.2(2.7)	97.1(2.6)	97.1(2.5)	96.9(2.6)	97.4(2.0)	97.3(2.2)	96.5(2.5)	97.3(1.9)	96.5(3.6)	96.5(2.5)	96.9(3.5)	96.8(3.6)	0.117
			Ud	89.8(4.3)	91.7(5.0)	92.5(3.8)	93.2(2.9)	94.9(2.7)	95.8(2.1)	95.4(2.7)	96.3(3.4)	96.3(2.2)	96.1(2.9)	95.4(3.5)	93.8(5.3)	91.4(6.3)	0.34
6	34.8	79.0(5.1)	Ov	76.7(7.0)	77.0(6.7)	77.3(7.5)	78.2(7.3)	77.3(8.2)	76.9(4.5)	79.1(5.3)	79.6(6.8)	75.7(6.9)	78.6(5.3)	78.3(7.2)	78.9(6.9)	79.4(5.2)	0.41
			Ud	60.8(10.0)	69.1(6.0)	69.2(5.9)	73.6(8.0)	77.9(5.2)	77.1(4.9)	80.2(6.1)	78.7(6.1)	78.8(6.2)	72.6(7.1)	63.7(10.5)	61.1(11.6)	57.7(9.0)	0.32
7	34.7	91.0(3.0)	Ov	93.1(2.7)	93.0(3.4)	92.7(3.4)	92.3(3.6)	92.1(4.0)	93.1(3.1)	91.5(3.7)	90.8(4.5)	88.8(2.4)	90.1(3.3)	89.2(3.8)	87.3(5.9)	86.6(5.8)	0.06
			Ud	50.0(0.0)	51.7(5.2)	53.4(6.0)	76.0(6.3)	78.3(5.8)	83.8(5.6)	87.8(5.4)	87.8(4.4)	84.9(4.5)	75.7(7.9)	69.6(9.4)	59.5(10.9)	52.4(5.0)	0.04
8	30.0	72.4(3.9)	Ov	69.9(6.1)	70.9(5.8)	69.1(4.8)	72.2(5.0)	70.9(4.0)	71.8(6.4)	72.5(5.9)	71.3(5.9)	72.6(4.0)	72.1(4.1)	73.2(4.3)	71.9(4.0)	72.4(4.5)	0.46
			Ud	50.0(0.0)	51.1(3.4)	50.0(0.0)	65.4(9.8)	70.4(7.9)	71.9(7.5)	70.4(5.9)	72.7(6.1)	72.8(4.0)	69.8(7.9)	50.0(0.0)	50.0(0.0)	50.0(0.0)	0.42
9	26.5	54.1(7.9)	Ov	60.5(7.2)	62.6(9.2)	62.9(7.8)	59.7(9.9)	62.5(10.0)	63.8(10.3)	64.1(12.2)	61.3(10.6)	64.3(14.7)	51.2(5.3)	54.9(13.2)	52.9(7.4)	52.6(8.7)	0.04
			Ud	50.0(0.0)	50.4(6.0)	52.1(6.7)	49.7(1.9)	54.9(7.8)	59.7(10.2)	61.7(11.4)	63.2(13.2)	61.0(14.4)	50.0(3.9)	50.4(1.4)	50.4(1.4)	49.7(1.0)	0.04
10	23.5	98.1(2.0)	Ov	97.7(1.8)	97.9(1.4)	97.9(1.1)	98.2(1.1)	98.1(2.1)	97.9(2.1)	97.8(2.0)	97.7(1.8)	97.8(1.5)	97.9(2.1)	97.9(2.2)	97.9(1.8)	97.7(2.3)	0.45
			Ud	75.6(7.0)	84.7(9.0)	87.5(7.8)	94.6(3.8)	95.3(2.8)	96.5(2.9)	97.3(2.5)	97.3(2.4)	97.5(1.8)	97.8(1.9)	97.5(2.3)	96.5(2.6)	94.0(3.3)	0.37
11	16.3	89.0(18.8)	Ov	91.2(18.9)	91.2(18.9)	91.2(18.9)	91.2(18.9)	91.2(18.9)	90.6(18.7)	90.6(18.7)	90.6(18.7)	90.3(18.5)	84.3(21.1)	87.9(18.2)	85.9(19.0)	92.6(10.7)	0.31
			Ud	55.3(14.2)	89.2(12.2)	88.2(10.1)	89.5(7.4)	85.0(16.5)	85.8(16.9)	87.7(17.8)	89.4(18.3)	87.5(18.1)	86.0(19.0)	92.6(12.7)	92.8(14.5)	86.4(18.3)	0.31
12	10.4	92.5(8.1)	Ov	94.9(5.5)	94.6(5.8)	94.3(6.8)	94.7(7.0)	94.3(7.2)	94.2(7.2)	93.1(6.7)	95.1(6.0)	95.2(6.2)	94.6(5.0)	92.7(8.3)	89.6(13.1)	93.3(9.0)	0.20
			Ud	69.8(13.9)	70.8(13.1)	72.9(10.1)	84.2(7.0)	85.8(5.6)	89.3(6.7)	93.1(4.2)	94.0(3.9)	93.0(2.9)	85.4(14.3)	94.3(4.4)	83.5(15.5)	80.3(20.0)	0.30
13	8.8	50.0(0.0)	Ov	66.2(22.8)	67.8(24.2)	67.0(23.3)	69.3(25.4)	66.5(26.5)	66.7(27.1)	63.4(17.5)	63.1(21.5)	55.8(22.8)	60.0(22.9)	49.2(1.9)	50.0(0.0)	50.0(0.0)	0.01
			Ud	50.0(0.0)	50.0(0.0)	50.8(2.6)	50.9(8.5)	60.5(6.6)	64.7(21.6)	59.3(29.7)	67.5(24.6)	64.6(28.1)	49.9(23.7)	50.0(0.0)	50.0(0.0)	50.0(0.0)	0.02
14	7.9	82.4(19.2)	Ov	85.4(14.7)	81.8(15.6)	82.2(15.6)	82.6(16.5)	82.9(15.3)	85.4(14.9)	85.6(15.5)	86.1(14.3)	85.0(17.0)	83.4(18.0)	84.1(17.6)	82.9(19.3)	85.8(15.5)	0.31
			Ud	50.0(0.0)	50.0(0.0)	56.5(6.5)	57.6(6.3)	63.9(19.3)	65.3(23.0)	67.1(16.8)	74.4(22.1)	69.3(22.0)	79.0(19.8)	83.4(18.0)	62.9(21.3)	51.7(5.3)	0.45

both means are the same. The smaller the p-value, the more evidence we have against H_0. A p-value lower than 0.05 indicates a 95% degree of confidence of rejecting H_0. Even though there are only a few differences at that significance level, some tendencies can be observed from these results.

A first observation from Table 2 is that random over-sampling performs better than random under-sampling. Out of 15 data sets used in this study, in only 3 of them (heart$_{(2)}$[2], pima$_{(6)}$ and new-thyroid$_{(11)}$) under-sampling performed slightly better than over-sampling. The reason is two-fold: over-sampling does not discard any cases and consequently it might not end up with a restricted set of examples which is unrepresentative of the underlying concept; and over-sampling increases the number of examples of the minority class, directly dealing with the problem of learning from the rare cases of this class.

Another observation is that changing the class distribution seems to be worth the effort. For random over-sampling, the best results obtained for each data set are higher than the performance obtained with the original class distribution. For random under-sampling, results seem to be less promising as for 5 (sonar$_{(1)}$, bupa$_{(3)}$, tic-tac-toe$_{(7)}$ and vehicle$_{(10)}$) of the 15 data sets, random under-sampling was not able to improve the performance obtained with the original class distribution.

As mentioned before, the best results obtained by over and under-sampling are shaded in gray. Most of these results are related to the most balanced class distributions, having a slight tendency to the left where, proportions are biased for the positive class.

Three of the most balanced distributions, *i*.e., 40%, 50% and 60% of positive class prevalence, concentrate exactly 7 (50%) of the best results obtained by random over-sampling. If we restrict the analysis to the balanced distribution, random over-sampling provided performance results slightly better than the balanced distribution in 13 out of the 15 data sets. Specifically for data sets haberman$_{(9)}$ and flag$_{(13)}$ which have less than 30% of positive examples and poor classification performance, and consequently, seem to suffer from the class imbalance problem, random over-sampling to the balanced distribution was able to improve the performance in two of them (haberman$_{(9)}$ and flag$_{(13)}$) with a statistical confidence of 95%[3].

It is important to note that the data sets german$_{(8)}$, vehicle$_{(10)}$, new-thyroid$_{(11)}$, ecoli$_{(12)}$ and glass$_{(14)}$ also have 30% or less positive class prevalence and do not seem to suffer from the class imbalance problem. For these data sets, balancing the class distribution did not improve the performance significantly. In addition, these data sets seem to confirm the hypothesis that the class imbalance does not hinder the performance of classifiers per se.

[2] From hereafter, we use a subscript number after a data set name in order to facilitate references to Table 2.

[3] For data set flag$_{(13)}$, the Student t-test p-value between the balanced distribution and the natural distribution is 2.42. It is not shown in Table 2 since the balanced distribution did not provide the best result for all considered distributions.

Class imbalance must be associated with other data characteristics such as the presence of within-class imbalance and small disjuncts [9] and data overlapping [11] in order to cause a loss in performance.

As a general recommendation given the results obtained in the experiments, random over-sampling seems to be a good starting point in order to deal with class imbalance. This method is straightforward to implement and considerably fast if compared with more sophisticated (heuristical) sampling methods. Over-sampling for the balanced distribution seems also to be a good first choice as AUC values near the balanced distribution are often the best.

As mentioned early, another point that is often cited in the literature is that over-sampling may lead to overfitting, due to the fact that random over-sampling makes exact copies of minority class examples. As results related to random over-sampling and overfitting are often reported using error rates as the basic performance measure, we believe that the conclusions reported might be due to the confusion of the classification criteria and the discrimination ability natural to the error rate measure. As a matter of fact, over-sampled data sets might produce classifiers with higher error rates than the ones induced from the original distribution. Since it is not possible to determine the appropriate configuration without knowing in advance the target distribution characteristics, it is not possible to confirm that over-sampling leads to overfitting. In fact, the apparent overfitting caused by over-sampling might be a shift into the classification threshold in the ROC curve.

Although for most of the sampled data sets it was not possible to identify significantly differences from the original distribution, this does not mean that the different sampling strategies or different proportions perform equally well, and that there is not any advantage in using one or another in a given situation. As stated earlier, this is due to the fact that the classifier with higher AUC values does not necessarily lead to the best classifier in the whole ROC space. The main advantage of using different sampling strategies relies on the fact that they could improve on different regions of the ROC space. In this sense, the sampling strategies and proportions could boost some rules that could be overwhelmed by imbalanced class distributions.

For instance, consider the ROC curves shown in Figure 1. This figure presents ROC graphs (averaged over the 10 folds using the vertical average method described in [8]) for the $\text{pima}_{(6)}$ data set. Furthermore, we have selected two curves which perform well in different parts of the ROC space. The selected curves are those generated from random under-sampled data sets with class distribution of 70% positive examples and random over-sampled data sets with 20% positive examples. Figure 1 shows that random under-sampling with 70% positive examples performs better in the range 0-50% of false positives, approximately. On the other hand, random over-sampled data sets with 20% positive examples outperforms random under-sampled data sets with 70% in the remainder of the ROC space. In other words, different sampling strategies and different class distribution may lead to improvements in different regions of the ROC space.

4 Concluding remarks

As long as learning algorithms use heuristics designed foroverall error rate minimization, it is natural to believe that these algorithms would be biased to perform better at classifying majority class examples than minority class ones, as the former is weighed more heavily when assessing the error rate.

However, it is possible to use learning algorithms that use basic heuristics insensitive to class distribution. One of these algorithms (a decision tree using DKM splitting criterion) is shown to be competitive to overall error minimization algorithms in various domains [5]. Furthermore, for some domains standard learning algorithms are able to perform quite well no matter how skewed the class distribution is, even if the applied algorithms are (at least indirectly) based on overall error rate minimization and therefore sensitive to class distribution. For these reasons, it is not fair to always associate the performance degradation in imbalanced domains to class imbalance.

Another point that is often cited as a drawback for learning in imbalanced domains is that, as the training set represents a sample drawn from the population, the examples belonging to the minority class might not represent all characteristics of the associated concept well. In this case, it is clear that the problem is the sampling strategy instead of the proportion of examples. If it were possible to improve the quality of the data sample, it would be possible to alleviate this problem.

Finally, it is worth noticing that generally there is a trade-off with respect to marginal error rates. This is to say that generally it is not possible to diminish the relative error rate of the minority class (false positive rate) without increasing the relative error rate of the majority class (false negative rate). Managing this trade-off introduces another variable in the scenario, namely misclassification costs. Although misclassification costs might be cast into a class (re)distribution by adjusting the expected class ratio [7], a complicating factor is that we do not generally know in advance the costs associated to

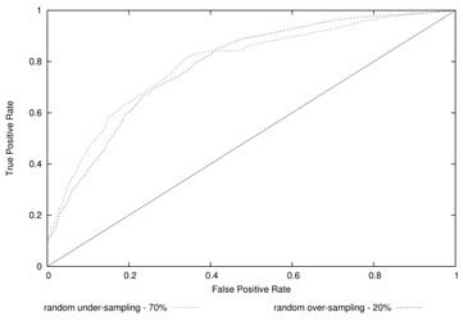


Fig. 1 Two ROC curves for the pima dataset, averaged over 10 folds. These ROC curves are those generated from random undersampled data sets with class distribution of 70% positive examples and random oversampled data sets with 20% positive examples.

each misclassification. ROC analysis is a method that analyses the performance of classifiers regardless of this trade-off by decoupling hit rates from error rates.

In order to investigate this matter in more depth, several further approaches might be taken. Firstly, it would be interesting to simulate different scenarios of class prior distributions and misclassification costs. This simulation could help us to identify in each situation which sampling strategy is preferred over another. Moreover, it is also interesting to apply some heuristic sampling methods, such as NCL [10] and SMOTE [3], as these sampling methods aim to overcome some limitations present in non-heuristic methods. Another interesting point is to empirically compare our method with algorithms insensitive to class skews. Finally, it would be interesting to further evaluate the induced models using different misclassification costs and class distribution scenarios. In the context of our experimental framework, it would be interesting to further evaluate how the sampling strategies modify the induced tree.

Acknowledgments This work was partially supported by the Brazilian Research Councils CNPq, CAPES, FAPESP and FPTI.

References

1. A. Asuncion, D.N.: UCI machine learning repository (2007). Http://www.ics.uci.edu/∼mlearn/MLRepository.html
2. Batista, G., Prati, R.C., Monard, M.C.: A Study of the Behaviour of Several Methods for Balance Machine Learning Training Data. SIGKDD Explorations **6**(1), 20–29 (2004)
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Over-sampling Technique. JAIR **16**, 321–357 (2002)
4. Cussens, J.: Bayes and Pseudo-Bayes Estimates of Conditional Probabilities and their Reliability. In: ECML'93, pp. 136–152 (1993)
5. Drummond, C., Holte, R.C.: Exploiting the Cost (In)Sensitivity of Decision Tree Splitting Criteria. In: ICML'2000, pp. 239–246 (2000)
6. Elkan, C.: Learning and Making Decisions When Costs and Probabilities are Both Unknown. In: KDD'01, pp. 204–213 (2001)
7. Elkan, C.: The Foudations of the Cost-sensitive Learning. In: IJCAI'01, pp. 973–978. Margan Kaufmann (2001)
8. Fawcett, T.: An introduction to ROC analysis. Pattern Recognition Letters **27**(8), 861–874 (2006)
9. Japkowicz, N.: Class Imabalances: Are we Focusing on the Right Issue? In: ICML'2003 Workshop on Learning from Imbalanced Data Sets (II) (2003)
10. Laurikkala, J.: Improving Identification of Difficult Small Classes by Balancing Class Distributions. Tech. Rep. A-2001-2, Univ. of Tampere, Finland (2001)
11. Prati, R.C., Batista, G.E.A.P.A., Monard, M.C.: Class Imbalance *versus* Class Overlapping: an Analysis of a Learning System Behavior. In: MICAI'04, pp. 312–321 (2004)
12. Provost, F.J., Fawcett, T.: Robust classification for imprecise environments. Machine Learning **42**(3), 203–231 (2001)
13. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann (1988)
14. Weiss, G.M., Provost, F.: Learning When Training Data are Costly: The Effect of Class Distribution on Tree Induction. JAIR **19**, 315–354 (2003)

Answer Extraction for Definition Questions using Information Gain and Machine Learning

Carmen Martínez-Gil[1] and A. López-López[2]

Abstract. Extracting nuggets (pieces of an answer) is a very important process in question answering systems, especially in the case of definition questions. Although there are advances in nugget extraction, the problem is finding some general and flexible patterns that allow producing as many useful definition nuggets as possible. Nowadays, patterns are obtained in manual or automatic way and then these patterns are matched against sentences. In contrast to the traditional form of working with patterns, we propose a method using information gain and machine learning instead of matching patterns. We classify the sentences as likely to contain nuggets or not. Also, we analyzed separately in a sentence the nuggets that are *left* and *right* of the target term (the term to define). We performed different experiments with the collections of questions from the TREC 2002, 2003 and 2004 and the F-measures obtained are comparable with the participating systems.

1 Introduction

Question Answering (QA) is a computer-based task that tries to improve the output generated by Information Retrieval (IR) systems. A definition question is a kind of question whose answer [12] is a complementary set of sentence fragments called nuggets.

After identifying the correct target term (the term to define) and context terms, we need to obtain useful and non redundant definition nuggets. Nowadays, patterns are obtained manually as surface patterns [7]. Also, patterns are very rigid,

[1] Carmen Martínez-Gil

Instituto Nacional de Astrofísica Óptica y Electrónica, Facultad de Ciencias de la Computación, Universidad de la Sierra Juárez,, email: carmen@inaoep.mx

[2] A. López-López

Instituto Nacional de Astrofísica Óptica y Electrónica, Luis Enrique Erro #1 Santa María Tonantzintla, 72840 Puebla, México, email: allopez@inaoep.mx

Please use the following format when citing this chapter:

Martínez-Gil, C. and López-López, A., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 141–50.

other case can be a soft pattern [4], even also extracted in an automatic way [5]. Then, once we have the patterns we apply a matching process to extract the nuggets. Finally, we need to perform a process to determine if these nuggets are part of the definition; where a common criterion employed is the repetition of the nugget.

According to the state of the art the F-measure in a pilot evaluation [12] for definition questions in 2002 is 0.688 using the nuggets set of author and 0.757 using the nuggets set of other with β=5. For the TREC 2003 [13] F-measure is 0.555 with β=5 and the TREC 2004 [14] F-measure is 0.460 with β=3.

In contrast to the traditional way to extract nuggets, we propose a method that uses two approaches: information gain and machine learning (ML), in particular Support Vector Machine (SVM), Random Forest (RF) and k-nearest-Neighbor (K-NN). We extract the sentence fragments to the *left* and *right* of the target term in an automatic way. These sentence fragments are obtained using a parser (Link Grammar) in the relevant sentences. Then, from parsed sentence we obtained four kinds of sentences fragments, noun phrase containing an appositive phrase, noun phrase containing two noun phrases separated by comma, embedded clauses, and main or subordinate clauses without considering embedded clauses. For the machine learning approach, we labeled with the correct tag, *positive* if the nugget is part of the definition and *negative* otherwise, to prepare the training set of a classifier. So, when we have a sentence fragment and we want to determine if it defines the target term, we apply the classifier.

For this task we work with the questions of the pilot evaluation of definition questions 2002, TREC 2003 and TREC 2004. First, we test each approach, i.e. frequencies, information gain and machine learning algorithms. Then, we combine the sentence fragments obtained with information gain and the sentence fragments labeled classified like *positive* by the ML algorithms.

The paper is organized as follows: next section describes the process to extract sentence fragments; Section 3 describes the approaches used and the method to retrieve only definition sentence fragments; Section 4 reports experimental results; some conclusions and directions for future work are presented in Section 5.

2 Sentence Fragments Extraction

Official definition of F-measure used in the TREC evaluations [12] is:

Let r # of vital nuggets returned in a response
a # of non-vital nuggets returned in a response
R total # of vital nuggets in the assessors' list
l # of non-whitespace characters in the entire answer string

Then

$$recall(\Re) = r / R \tag{1}$$

$$allowance(\alpha) = 100 \times (r + a) \tag{2}$$

$$precision(\mathrm{P}) = \begin{cases} 1 & if \quad l < \alpha \\ 1 - \dfrac{l - \alpha}{l} & otherwise \end{cases} \tag{3}$$

$$\text{Finally, the } F(\beta = 3) = \frac{(\beta^2 + 1) \times \mathrm{P} \times \Re}{\beta^2 \times \mathrm{P} + \Re} \tag{4}$$

So, a reason to extract sentence fragments is that we need to retrieve only the most important information from relevant sentences. Other reason to extract short sentence fragments is related to the performance F-measure applied to definition systems in the TREC evaluation; this measure combines the recall and precision of the system. The precision is based on length (in non-white-space characters) used as an approximation to nugget precision. The length-based measure starts from an initial allowance of 100 characters for each (vital or no-vital) nugget matched. Otherwise, the measure value decreases as the length the sentence fragment increases.

We use Lucene [15] system to extract candidate paragraphs from the AQUAINT Corpus of English News Text. From these candidate paragraphs we extract the relevant sentences, i.e. the sentences that contain the target term. Then, to extract sentence fragments we proposed the following process:

1) **Parse the sentences.** Since we need to obtain information segments (phrases or clauses) from a sentence, the relevant sentences were parsed with Link Grammar [6]. We replace the target by the label **SCHTERM**. For example the sentence for the target term **Carlos the Jackal**:

   The man known as **Carlos the Jackal** has ended a hunger strike after 20 days at the request of a radical Palestinian leader, his lawyer said Monday.

   The Link Grammar produces:

```
[S [S [NP [NP The man NP] [VP known [PP as [NP
SCHTERM NP] PP] VP] NP] [VP has [VP ended [NP a hun-
ger strike NP] [PP after [NP 20 days NP] PP] [PP at
[NP [NP the request NP] [PP of [NP a radical Pales-
tinian leader NP] PP] NP] PP] VP] VP] S] , [NP his
lawyer NP] [VP said [NP Monday NP] . VP] S]
```

2) **Resolve co-references.** We want to obtain main clauses without embedded clauses or only embedded clauses, so we need to resolve the co-reference, otherwise important information can be lost. To resolve co-reference the relative pronouns WHNP are replaced with the noun phrase preceding it.

3) **Obtain sentence fragments.** An information nugget or an atomic piece of information can be a phrase or a clause. We analyzed the sentences parsed with Link Grammar and we have identified four kinds of sentence fragments directly

related to the target with a high possibility that their information define the target:

a) Noun phrase (NP) containing an appositive phrase.
b) Noun phrase (NP) containing two noun phrases separated by comma [NP, NP].
c) Embedded clauses (SBAR).
d) Main or subordinate clauses (S) without considering embedded clauses.

To retrieve the four kinds of sentence fragments we analyze the tree following this procedure:

I. Looking for the nodes which contain the target, in our case the label SCHTERM.
II. Find the initial node of the sentence fragment. The process analyzes the path from the node with the SCHTERM label towards the root node. The process stops when a NP with appositive phrase, NP with [NP, NP], an embedded clause SBAR, or a clause S is reached.
III. Retrieve the sentence fragment without embedded clauses.
IV. Mark as visited the parent node of the second phrase. In case [NP1, NP2] mark as visited the parent node of NP2. For appositive phrase, SBAR or S, the second phrase can be NP, VP or PP.

The steps II – IV are repeated for the same node with a SCHTERM label until a visited node is found in the path to the node towards the root node or the root node is reached. Also the steps II – IV are repeated for each node found in step I.

The next module of our definition question system selects definition sentence fragments. In order to select only definition nuggets from all of sentence fragments, we analyze separately, the information that is to the left of SCHTERM and the information that is to the right of SCHTERM, so we form two data sets.

Now, we present some sentence fragments of two sets obtained using the process for the target term **Carlos the Jackal**:

```
Right sentence fragments
SCHTERM , a Venezuelan serving a life sentence in a French prison
SCHTERM , nickname for Venezuelan born Ilich Ramirez Sanchez
SCHTERM , is serving a life sentence in France for murder
SCHTERM as a comrade in arms in the same unnamed cause
SCHTERM refused food and water for a sixth full day
SCHTERM , the terrorist imprisoned in France

Left sentence fragments
the friendly letter Chavez wrote recently to the terrorist
SCHTERM
The defense lawyer for the convicted terrorist known as SCHTERM
he was harassed by convicted international terrorist SCHTERM
an accused terrorist and a former accomplice of SCHTERM
Ilich Ramirez Sanchez , the terrorist known as SCHTERM
Ilich Ramirez Sanchez , the man known as SCHTERM
```

Analyzing separately the sentence fragments before and after the target term is an advantage since in many candidate sentences only one part contains information that defines the target term.

3 Nuggets Selection

In order to obtain only the informative nuggets from the *left* and *right* sentence fragments we use two approaches, one using statistical methods and the other using machine learning algorithms. In the statistical methods we assess the information gain of each fragment and simple frequencies. For the latter we only obtained word frequencies for the sake of comparison. We describe information gain and the machine learning algorithms.

3.1 Information Gain

The information gain [2] for each word or term *l* is obtained using the following definition:

Given a set of sentence fragments *D*, the entropy *H* of *D* is:

$$H(D) \equiv \sum_{i=1}^{c} - p_i \log_2 p_i \tag{5}$$

Where *Pi* is the probability of *i* word and *c* is the size of the vocabulary. Now, for each term *l*. Let D^+ be the subset of sentence fragments of *D* containing *l* and D^- its complement. The information gain of *l*. *IG(l),* is defined by

$$IG(l) = H(D) - \left[\frac{|D^+|}{|D|} H(D^+) + \frac{|D^-|}{|D|} H(D^-) \right] \tag{6}$$

3.2 Machine Learning Algorithms

The other approach to determine if a sentence fragment is part of the definition is using a machine learning algorithm, if it is labeled like positive, then is part of a definition sentence. The ML algorithms that we used are Support Vector Machine, Random Forest, and k-Nearest-Neighbors. We describe briefly each algorithm in the following sections.

Support Vector Machine

The Support Vector Machine (SVM) is a classification technique developed by Vapnik [3], [11]. The method conceptually implements the idea that input vectors

are no-linearly mapped to a very high-dimension feature space. In this feature space a linear decision surface is constructed. Special properties of the decision surface ensure high generalization ability of the learning machine. The main idea behind the technique is to separate the classes with a surface that maximizes the margin between them. SVM is based on the Structural Risk Minimization (SRM) principle [11] from computational learning theory. We used a polynomial kernel to perform our experiments.

Random Forest

Random Forest [1] is a classifier that consists of several decision trees. The method uses Breiman's bagging idea and Ho's random subspace method [8] to construct a collection of decision trees with controlled variations. Random forests are a combination of tree predictors such that the tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.

K-Nearest-Neighbor

K-Nearest-Neighbor (K-NN) belongs to the family of instance-based learning algorithms. These methods simply store the training examples and when a new query instance is presented to be classified; its relationship to the previously stored examples is examined in order to assign a target function value. A more detailed description of this algorithm can be found in [9]. In this work, we use distance-weighted K-NN.

3.3 Method to Select Nuggets

To obtain informative nuggets we combine two processes, one using information gain and the other using machine learning algorithms. The process that uses information gain is the following:

I) Obtain the vocabulary of all the sentence fragments (*left* and *right* sets).
II) Obtain the information gain for each word of the vocabulary using the definition of section 3.1.
III) Using the value of the information gain of each word (except stop words), calculate the sum of each sentence fragment.
IV) Rank the sentences fragments according to the value of the sum.
V) Eliminate redundant sentence fragments.

To eliminate redundancy, we compare pairs (*X, Y*) of sentence fragments using the following steps:

a) Obtain the word vector without empty words for each sentence fragment.
b) Find the number of identical words between the two Sentence Fragments *SF*.

c) If $\frac{SF}{|X|} \geq \frac{2}{3}$ or $\frac{SF}{|Y|} \geq \frac{2}{3}$, remove the sentence fragment with lower sum of information gains of the vector. We tested others thresholds but with 2/3 we obtained the better results.

To illustrative the process to eliminate redundancy, we present the following sentence fragments for the target **Carlos the Jackal,** with their corresponding sums:

```
2.290 nickname for Venezuelan born Ilich Ramirez Sanchez
2.221 Ilich Ramirez Sanchez , the Venezuelan born former guer-
rilla
2.157 Ilich Ramirez Sanchez , the terrorist
1.930 Ilich Ramirez Sanchez , the man
1.528 Illich Ramirez Sanchez
```

If we compare the first and the second sentences, the result of the step a) is:

```
[nickname, Venezuelan, born, Ilich, Ramirez, Sanchez]
[Ilich, Ramirez, Sanchez, Venezuelan, born, former, guerrilla]
```

In the step b) we obtained that SW=5.

Finally, in the step c) we remove the second sentence fragment since it has a lower sum of information gains. Applying the procedure with the other sentence fragments, the result is that we keep only the first:

2.290 nickname for Venezuelan born Ilich Ramirez Sanchez

For the machine learning algorithms we apply the following process. From the AQUAINT Corpus and following the process described in the section 2, we obtained the sentence fragments to form two training sets for the three learning algorithms. The *left* set contains 2982 examples and the *right* set contains 3681 examples. The sets were formed with a ratio of 1:3 between positive and negative examples in order to have balanced sets. One sentence fragment was labeled as *positive* if it contains information of a vital or no vital nugget and *negative* otherwise. The sentence fragments were tagged with POS [10]. Then, we maintain the two words closer to the target term and the following five tags, so a window of seven words and tags is obtained. We tested others combinations likes all labels POS or maintain the word closer to the target but the best result was obtained maintain the two words closer.

An illustrative example to obtain the training set for the target **Christopher Reeve**, using only three sentences fragments, is the following:

Right set of sentence fragments

```
SCHTERM is paralyzed from a spinal cord injury in a rid-
ing accident
SCHTERM, the actor confined to a wheelchair from a
horseback riding accident
SCHTERM told a 6 year old girl paralyzed in an amusement
park accident
```

Sentence fragments tagged with POS

```
SCHTERM/NNP is/VBZ paralyzed/VBN from/IN a/DT spinal/JJ
cord/NN injury/NN in/IN a/DT riding/VBG accident/NN
```

```
SCHTERM/NNP ,/, the/DT actor/NN confined/VBD to/TO a/DT
wheelchair/NN from/IN a/DT horseback/NN riding/VBG acci-
dent/NN
SCHTERM/NNP told/VBD a/DT 6/CD year/NN old/JJ girl/NN
paralyzed/VBN in/IN an/DT amusement/NN park/NN acci-
dent/NN
```

Final coding for training set

```
is, paralyzed, IN, DT, JJ, NN, NN, POSITIVE
COMMA, the, NN, VDB, TO, DT, NN, POSITIVE
told, a, CD, NN, JJ, NN, VBN, POSITIVE
```

4 Experiments Results

We performed experiments with three sets of definition question, the questions from the pilot evaluation 2002, TREC 2003 and TREC 2004. (We did not compare our results with the collections of the TREC 2005 and 2006 since in these years the list of nuggets was not readily available). First we test each approach, i.e. frequencies, information gain and the machine learning algorithms. For the latter approach we used the training set described in the section 3.3 but excluding from the training set the collection on evaluation. Then, we combine the sentence fragments obtained with information gain and the sentence fragments classified like *positive* by the machine learning algorithms.

Values of the F-measure are shown in the figure 1 and Freq is the baseline. In every set of questions, information gain obtained higher F-measure than simple frequencies and machine learning algorithms. But the best value of the F-measure is obtained when we combined information gain with the machine learning algorithms, since the two approaches are complementary, the first approach obtained the most frequent sentence fragments and the second approach retrieves the information that has implicitly or explicitly a definition pattern.

It is important to note that with the collection 2002 there are two set of nuggets AUTHOR and OTHER. We compare the output of our system (labeled SysDefQuestions) with the set's AUTHOR nuggets. Figure 2 shows the comparison of F-measure values obtained in the pilot evaluation version of definition questions using the AUTHOR set of nuggets [12]. The figure 4 shows the comparison of F-measure values obtained in the TREC 2003 [13]. Finally, in the figure 5 we present the comparison of F-measure values obtained in the TREC 2004[14].

	Freq	IG	ML	IG+ML
2002	0,64	0,733	0,613	0,738
2003	0,368	0,425	0,4	0,443
2004	0,227	0,289	0,278	0,303

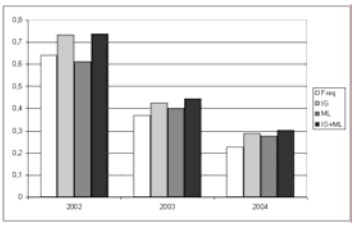


Fig. 1. Comparison of the F-measures obtained with Frequencies Freq, information gain IG, machine learning algorithms ML, and the combination of IG with ML.

Pilot 2002		
1	**SysDefQuestions**	**0,715**
2	F	0,688
3	A	0,606
4	D	0,568
5	G	0,562
6	E	0,555
7	B	0,467
8	C	0,349
9	H	0,33

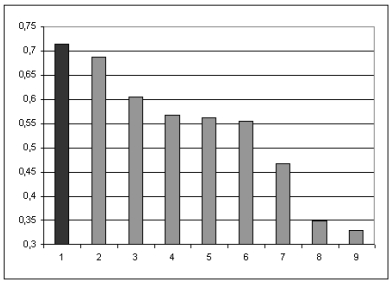


Fig. 2. Comparison of F-measure values of pilot evaluation of definition questions using the AUTHOR list of nuggets.

TREC 2003		
1	**SysDefQuestions**	**0,563**
2	BBN	0,555
3	National University of Singapore	0,473
4	University of Southern Calif. ISI	0,461
5	Language Computer Corp.	0,442
6	Univ. Of Colorado/Columbia Univ.	0,338
7	ITC-irst	0,318
8	Univ. Of Amsterdam	0,315
9	MIT	0,309
10	Univ. Of Sheffield	0,236
11	Univ. Of Iowa	0,231
12	Carnegie Mellon University	0,216
13	Fundan University	0,192
14	Univ. Of Pisa	0,185
15	IBM Research	0,177
16	NTT Communication Science Labs	0,169

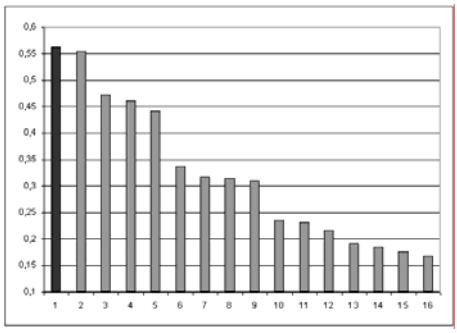


Fig. 3. Comparison of F-measure values of TREC 2003.

TREC 2004		
1	National Univ. Of Singapore	0,46
2	**SysDefQuestions**	**0,412**
3	Fudan University	0,404
4	National Security Agency	0,376
5	University of Sheffield	0,321
6	University of North Texas	0,307
7	IBM Research	0,285
8	Korea University	0,247
9	Language Computer Corp.	0,24
10	CL Research	0,239
11	Saarland University	0,211

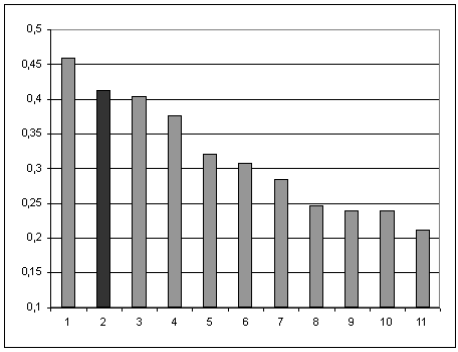


Fig. 4. Comparison of F-measure values of TREC 2004.

From two sets of definition questions, we can observe that our system SysDefQuestions retrieves most of the definition sentence fragments. For the set of definition questions of TREC 2004 the F-measure of our system is competitive when compared to the participating systems.

5 Conclusions and Future Works

We have presented a method to extract definition sentence fragments called nuggets in an automatic and flexible way and the results obtained are comparable with the participating systems in the TREC. The sentence fragments obtained with the process presented are acceptable since these contain only the information directly related to the target. Other advantage is that these sentence fragments present a short length, and this improves the precision of our definition question system.

We are planning to categorize the targets in three classes: ORGANIZATIONS, PERSON and ENTITIES and then train three different classifiers.

Acknowledgments: The first author was supported by scholarship 157233 granted by CONACyt, while second author was partially supported by SNI, Mexico.

References

1. Breiman, L.: Random Forest. Machine Learning 45 (1), (2001) 5-32.
2. Carmel, D., Farchi, E., Petruschka, Y., and Soffer, A.: Automatic Query refinement using lexical affinities with maximal information gain. *SIGIR* (2002): 283-290.
3. Cortes, C. and Vapnik, V.: Support Vector Networks. Machine Learning. (1995) 20:1-25.
4. Cui, H., Kan, M. Chua, T. and Xiao, J.: A Comparative Study on Sentence Retrieval for Definitional Questions Answering. SIGIR Workshop on Information Retrieval for Question Answering (IR4QA), (2004) 90-99.
5. Denicia-Carral, C., Montes-y-Gómez, M., and Villaseñor-Pineda, L.: A Text Mining Approach for Definition Question Answering. 5th International Conference on Natural Language Processing, Fin Tal. Lecture Notes in Artificial Intelligence, Springer (2006).
6. Grinberg, D., Lafferty, J., and Sleator, D.: A robust parsing algorithm for link grammars. Carnegie Mellon University Computer Science technical report CMU-CS-95-125, and *Proceedings of the Fourth International Workshop on Parsing Technologies*, Prague, September, (1995).
7. Hildebranddt, W., Katz, B. and Lin, J.: Answering Definition Question Using Multiple Knowledge Sources. In Proceeding of HLT/NAACL, Boston (2004) 49-56.
8. Ho, T.: The Random Subspace Method for Constructing Decision Forests. IEEE Trans. on Pattern Analysis and Machine Intelligence 20 (8), (1998) 832-844.
9. Mitchell, T.: *Machine Learning*. McGraw-Hill. (1997).
10. Toutanova, K., Klein, D., Manning, C., and Singer, Y.: Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. *In Proceedings of HLT-NAACL* (2003): 252-259.
11. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York. (1995).
12. Voohees, E.: Evaluating Answering to Definition Questions. NIST (2003) 1-3.
13. Voorhees, E.: Overview of the TREC 2003 Question Answering Track. *NIST* (2003): 54-68.
14. Voorhees, E.: Overview of the TREC 2004 Question Answering Track. *NIST* (2004): 12-20.
http://lucene.apache.org/java/docs/

Batch Reinforcement Learning for Controlling a Mobile Wheeled Pendulum Robot

Andrea Bonarini, Claudio Caccia, Alessandro Lazaric, and Marcello Restelli

Abstract In this paper we present an application of Reinforcement Learning (RL) methods in the field of robot control. The main objective is to analyze the behavior of batch RL algorithms when applied to a mobile robot of the kind called *Mobile Wheeled Pendulum* (MWP). In this paper we focus on the common problem in classical control theory of following a reference state (e.g., position set point) and try to solve it by RL. In this case, the state space of the robot has one more dimension, in order to represent the desired variable state, while the cost function is evaluated considering the difference between the state and the reference. Within this framework some interesting aspects arise, like the ability of the RL algorithm to generalize to reference points never considered during the training phase. The performance of the learning method has been empirically analyzed and, when possible, compared to a classic control algorithm, namely linear quadratic optimal control (LQR).

1 Introduction

This paper is about the application of Reinforcement Learning (RL) methods [10] in the field of robot control. To achieve optimal performance, many feedback control techniques (e.g., PID, direct pole placement, optimal control, etc.) generally require very accurate models of the dynamics of the robot and of its interaction with the

Andrea Bonarini, Alessandro Lazaric, Marcello Restelli
Dept. of Electronics and Information
Politecnico di Milano
Piazza Leonardo da Vinci 32, I-20133 Milan, Italy
e-mail: {bonarini,lazaric,restelli}@elet.polimi.it

Claudio Caccia
Dept. of Informatics, Systems and Communication
Università degli Studi di Milano - Bicocca
Viale Sarca 336, I-20126 Milan, Italy e-mail: caccia@disco.unimib.it

Please use the following format when citing this chapter:

Bonarini, A., Caccia, C., Lazaric, A. and Restelli, M., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 151–60.

surrounding environment, which is often infeasible in many real situations. Using traditional RL techniques (e.g., Q-learning), a robot can learn the optimal control policy by directly interacting with the environment, without knowing any model in advance. On the other hand, collecting data through direct interaction is a very long process when real robots are considered. Furthermore, RL algorithms are typically used to solve single-task problems such as balancing an inverted pendulum, driving the robot to a goal state, or learning to reach a given set point. This implies that every time the goal changes the learning process must restart from scratch, thus making infeasible to cope with a common problem in control theory like following a continuously changing reference point (e.g., position or speed set points). To face this class of problems, the state space of the problem needs to be enlarged by adding a new variable to represent the desired state, while the cost function can be defined on the error signal (i.e., the distance between the current state and the desired one).

In this paper, we propose to use *fitted Q-iteration* algorithms [6, 9, 2], i.e., batch algorithms that decompose the original RL problem into a sequence of supervised problems defined on a set of samples of state transitions. Since the value of the reference state does not affect the transition model, but only the reward function, using a batch approach allows to reuse the same set of transition samples to train the controller for different values of the reference state thus reducing the time of direct interaction with the environment. Within this framework some interesting aspects arise, for example the ability of the RL algorithm to generalize to reference points never seen during the training phase. The experimental activity has been carried on using a model of a mobile robot of the kind called Mobile Wheeled Pendulum. We experimentally evaluate batch RL algorithms using different function-approximation techniques, and compare their accuracies when following a given angle profile.

The rest of the paper is organized as follows: next section briefly motivates the use of batch RL algorithms and reviews the main state-of-the-art approaches. Section 3 describes the dynamic model of the robot Tilty used in the experiments. In Section 4 we describe how to collect data and how to train batch RL algorithms to build automatic controllers, and in Section 5 we show the results obtained using both neural networks and extra-randomized trees. Section 6 draws conclusions and proposes new directions for future research.

2 Batch Reinforcement Learning

In reinforcement learning [10] problems, an agent interacts with an unknown environment. At each time step, the agent observes the *state*, takes an *action*, and receives a *reward*. The goal is to learn a *policy* (i.e., a mapping from states to actions) that maximizes the long-term return. An RL problem can be modeled as a Markov Decision Process (MDP) defined by a tuple $\langle \mathscr{S}, \mathscr{A}, \mathscr{T}, \mathscr{R}, \gamma \rangle$, where $\mathscr{S}$ is the set of states, $\mathscr{A}(s)$ is the set of actions available in state s, $\mathscr{T} : \mathscr{S} \times \mathscr{A} \times \mathscr{S} \to [0,1]$ is a transition distribution that specifies the probability of observing a certain state after taking a given action in a given state, $\mathscr{R} : \mathscr{S} \times \mathscr{A} \to \mathbb{R}$ is a reward

function that specifies the instantaneous reward when taking a given action in a given state, and $\gamma \in [0,1]$ is a discount factor. The policy of an agent is characterized by a probability distribution $\pi : \mathscr{S} \times \mathscr{A} \rightarrow [0,1]$ that specifies the probability of taking a certain action in a given state. The utility of taking action a in state s and following a policy π thereafter is formalized by the action-value function $Q^{\pi}(s,a) = E\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t | s = s_1, a = a_1, \pi\right]$, where $r_1 = \mathscr{R}(s,a)$. RL approaches aim at learning the policy that maximizes the action-value function in each state. The optimal action-value function is the solution of the Bellman equation: $Q^*(s,a) = R(s,a) + \gamma \sum_{s'} \mathscr{T}(s,a,s') \max_{a'} Q^*(s',a')$. The optimal policy $\pi^*(\cdot,\cdot)$ takes in each state the action with the highest utility.

Temporal Difference algorithms [10] allow the computation of $Q^*(s,a)$ directly interacting with the environment with a trial-and-error process. Given the tuple $\langle s_t, a_t, r_t, s_{t+1} \rangle$ (experience made by the agent), at each step, action values may be estimated by online algorithms, such as Q-learning [10], whose update rule is: $Q(s_t,a_t) \leftarrow (1-\alpha) Q(s_t,a_t) + \alpha (r_t + \gamma \max_{a'} Q(s_{t+1},a'))$:$\alpha \in [0,1]$ learning rate.

RL has proven to be an effective approach to solve finite MDPs. On the other hand, using RL techniques in robotic and control applications rises several difficulties. Since state and action spaces are high-dimensional and continuous, the value function cannot be represented by means of tabular approaches, but function-approximation techniques are required. Despite some successful applications, coupling function approximation with *online* RL algorithms can lead to oscillatory behaviors or even to divergence [1]. The reason for this is that, unlike in the supervised case, in RL we cannot sample from the target function, and the training samples depend on the function approximator itself. Recently, several studies have focused on developing *batch* RL algorithms. While in online learning the agent modifies its control policy at each time step according to the experience gathered from the environment, the batch approach aims at determining the best control policy given a set of experience samples $\langle s_t, a_t, r_t, s_{t+1} \rangle$ previously collected by the agent following a given sampling strategy. In particular, good results have been achieved by *fitted Q-iteration* algorithms derived from the *fitted value iteration* approach [3]. The idea is to reformulate the RL problem as a sequence of supervised learning problems. Given the dataset, in the first iteration of the algorithm, for each tuple $\langle s_i, a_i, r_i, s_i' \rangle$, the corresponding training pair is set to $(s_i, a_i) \rightarrow r_i$, and the goal is to approximate the expected immediate reward $Q_1(s,a) = E[\mathscr{R}(s_t,a_t) | s_t = s, a_t = a]$.

The second iteration, based on the approximation of the Q_1-function, extends the optimization horizon one step further: $Q_2(s,a) = \mathscr{R}(s,a) + \gamma \max_{a'} \hat{Q}_1(s',a')$. At the N^{th} iteration, using the approximation of the Q_{N-1}-function, we can compute an approximation of the optimal action-value function at horizon N.

The batch approach allows to use any regression algorithm, and not only parametric function approximators as happens for stochastic approximation algorithms. Several studies have reported very good results with a wide range of approximation techniques: kernel-based regressors [6], tree-based regressors [2], and neural networks [9]. All these works show how batch mode RL algorithms allow to effectively exploit the information contained in the collected samples, so that, even using small datasets, very good performances can be achieved. The size of the dataset is

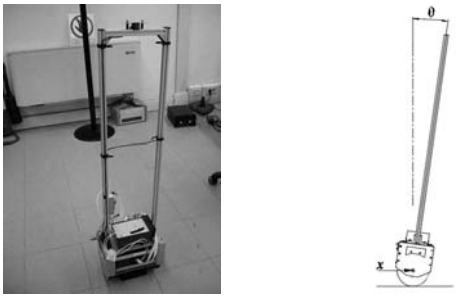


Fig. 1 The MWP robot *Tilty* and its degrees of freedom

a key factor for robotic applications, since collecting a large amount of data with real robots may be expensive and dangerous. As shown in [9], it is possible to solve simple control problems, such as balancing a pole, with a dataset obtained by performing random actions for a few minutes.

In this paper, we study the problem of controlling a mobile wheeled pendulum to follow a time-dependent set point. As we will explain in Section 5, fitted Q-iteration algorithms are well-suited to cope with this class of problems and we will show the results achieved with different function approximators.

3 The Robot: *Tilty*

The mobile wheeled inverted pendulum robot named *Tilty* has been considered for the tests. Tilty has an aluminum frame, a pair of wheels on the same axis connected to a DC motor each, batteries and a programmable drive. The robot structure is represented in Figure 1. The system has two types of sensors onboard: encoders on motors and a dual-axis accelerometer.

3.1 Dynamical Model

The first analysis consists in the study of the equations that describe the dynamics of the robot. In our case the model describing *Tilty* is non-linear: i.e., the system cannot be described by a system of equations of the form $\dot{x}(t) = \mathbf{A} \cdot x(t) + \mathbf{B} \cdot u(t)$, with $\mathbf{A}$ and $\mathbf{B}$ constant matrices, but by the general form $\dot{x} = f(x(t), u(t))$.

For mechanical or dynamical models, a common way to obtain such non-linear equations is to use the *Lagrange equations* [4], [8]:

$$\frac{d}{dt}\left(\frac{\partial \mathbf{T}}{\partial \dot{q}}\right) - \frac{\partial \mathbf{T}}{\partial q} - \frac{\partial \mathbf{V}}{\partial q} = \tau, \tag{1}$$

where q is the generic variable describing the pose of the system (i.e., degree of freedom), $\mathbf{T}$ is the kinetic energy, $\mathbf{V}$ is the potential energy, and τ represents the generalized force acting on the system. *Tilty* is described by two degrees of freedom when considering a linear motion (see Figure 1): the position of the center of the wheels x, and the angle ϑ between the pole and the vertical axis, while the input is represented by the motor torque $\mathbf{C}$ acting between the motors and the pendulum.

The equations for the kinetic and potential energy of the system are:

$$\begin{cases} \mathbf{T}(x,\vartheta) = \frac{1}{2}\cdot M_{tot}\dot{x}^2 + \frac{1}{2}\cdot J_{tot}\dot{\vartheta}^2 + H_{tot}\cdot\cos(\vartheta)\dot{\vartheta}\dot{x} \\ \mathbf{V}(x,\vartheta) = -H_{tot}(1-cos(\vartheta))\cdot g \end{cases} \tag{2}$$

where $M_{tot} = \sum m_i$ is the total mass of the system, $J_{tot} = \sum J_i + \sum m_i \cdot d_i^2$ is the moment of inertia w.r.t. the wheel axis, and $H_{tot} = \sum m_i \cdot d_i$ is the first order moment.

In order to calculate the generalized forces τ in (1), the virtual work of acting forces has to be determined: $\tau_\mathbf{i} = \frac{\delta^* L}{\partial q_i}$ with $\delta^* L = 2C \cdot \left(\frac{\delta^* x}{R_{wheel}} - \delta^* \vartheta\right)$. Where C is the motor torque acting on wheels. Solving the equations in (1) with the expressions in (2) brings to the following system of equations:

$$\begin{cases} M_{tot}\ddot{x} + H_{tot}\cos(\vartheta)\cdot\ddot{\vartheta} - H_{tot}\sin(\vartheta)\cdot\dot{\vartheta}^2 = \frac{2C}{R_{wheel}} \\ J_{tot}\ddot{\vartheta} - H_{tot}\cos(\vartheta)\cdot\ddot{x} - H_{tot}\sin(\vartheta)\cdot g = -2C \end{cases} \tag{3}$$

which represents the non-linear dynamics of *Tilty*. The system in (3) can be rearranged in the form:

$$\left\{\ddot{x},\dot{\vartheta},\ddot{\vartheta}\right\}' = \left[\mathbf{A}\left(\vartheta,\dot{\vartheta}\right)\right]\cdot\left\{\dot{x},\vartheta,\dot{\vartheta}\right\}' + \left\{\mathbf{B}\left(\vartheta\right)\right\}\cdot C \tag{4}$$

The matrix $\mathbf{A}$ and vector $\mathbf{B}$ in (4) are the state space representation of the system's dynamics.

3.2 Design of the controller

The system needs a regulator able to keep it in equilibrium. We developed a regulator of the kind LQR (linear quadratic regulator), obtained by optimal control theory [5]. Optimal control finds a regulator able to give the best performance with respect to a specific measure of the performance itself. LQR procedure is interesting because it allows to minimize the cost functional of the following equation (5) giving stable controllers and because it is applicable to Multi-Input Multi-Output (MIMO) systems.

$$\mathbf{J} = \int_0^\infty \left(x'\cdot\tilde{\mathbf{Q}}\cdot x + u'\cdot\tilde{\mathbf{R}}\cdot u\right)dt, \tag{5}$$

where x is the state of the system, u is the input variable, $\tilde{\mathbf{Q}}$ and $\tilde{\mathbf{R}}$ are weight matrices for state variables and actions. The values $\tilde{\mathbf{Q}}$ and $\tilde{\mathbf{R}}$ have been chosen to optimize the system response in the conditions of the experiments. The feedback control law that

minimizes the value of the cost is $\mathbf{u} = -K \cdot \mathbf{x}$. This feedback law is determined by solving the *Riccati Equation* [5]. Therefore, it is necessary to linearize (4) around an equilibrium point, so that matrix **A** and vector **B** become state independent. This approach gives controllers with good behavior, but are heavily dependent on the model of the system: it is necessary to describe exactly the geometry and the dynamics and assume that the linear approximation is good. RL methods, proposed here, do not need any prior knowledge about the system, so appear to be interesting when the model is strongly non-linear or when its parameters can be hardly estimated.

3.3 Simulation of the dynamics

The behavior of the controlled system has been simulated as accurately as possible, considering, among others, the following aspects:

- control frequency of 50Hz,
- the robot inclination ϑ is available by reading the output of the 2-axis accelerometer, obtaining the inclination by comparing the two data,
- the robot angular velocity $\dot{\vartheta}$ is not directly available by sensor reading, so its value is determined by means of a reduced observer block.

The model described here has been used to gather all the data used to apply batch RL algorithms and to compare the performance of the LQR controller with the policy determined by the learning algorithms.

4 Experimental Settings

The application of RL methods requires the agent to interact with the environment in order to learn the optimal policy. The interaction can be direct (on-line) or indirect (off-line). In the first case, the agent itself chooses the action based on what it has learned until then, and the policy (the *Q-function*) is estimated progressively. In the second case, the policy update is done in a batch fashion. Rather than choosing actions based on a policy, the agent observes state transitions due to actions externally determined. On the basis of the whole dataset, the optimal policy is computed. Therefore, the first step in batch RL methods consists in collecting samples $\langle s, a, r, s' \rangle$.

4.1 Data collection

The kind of raw data needed for training is made of tuples $(\mathbf{s}, a, \mathbf{s}')$, where $\mathbf{s} = \{\dot{x}, \vartheta, \dot{\vartheta}\}$ is the present state of the robot, a is the torque $\mathbf{C}$ applied, and $\mathbf{s}' =$

$\{\dot{x}', \vartheta', \dot{\vartheta}'\}$ is the next state reached from state $\mathbf{s}$ when $\mathbf{C}$ is applied. The whole dataset is composed by seven-tuple samples: $\left\{\dot{x}, \vartheta, \dot{\vartheta}, \mathbf{C}, \dot{x}', \vartheta', \dot{\vartheta}'\right\}$.

Each sample represents a Markovian transition. Using model-free RL algorithms, it is not required a priori knowledge about system dynamics, since it can be implicitly inferred by the samples obtained through direct interaction with the real robot.

In our approach, we consider the dynamical model described in Section 3.1. The model is initialized with a random state defined by the vector $[0, \vartheta_{in}, 0]'$, with ϑ_{in} varying in the range of ± 0.3rad. A random motor torque uniformly distributed in $\pm C_{max} = 7.6\ Nm$ is then applied to the system at frequency of 50Hz and the sequence of the states reached is collected with the same frequency. When the system reaches dangerous conditions (i.e., $|\vartheta| \geq 0.5$rad), the simulation is stopped and the system is initialized again. All the experiments have been carried out using datasets with $1,000$, $3,000$, and $5,000$ samples, that correspond, respectively, to $20s$, $60s$, and $100s$ of training in real time. During the phase of data collection no reference value is considered. Then, we have considered angular references by adding two values to each sample, thus obtaining the input vector of the training set: $\left\{\dot{x}, \vartheta, \dot{\vartheta}, \vartheta_{ref}, \mathbf{C}, \dot{x}', \vartheta', \dot{\vartheta}', \vartheta'_{ref}\right\}$. We made the assumption that the reference varies slowly w.r.t. the frequency of data collection, so that it can be considered constant during a single transition ($\vartheta'_{ref} = \vartheta_{ref}$). We consider the following set of reference values: $\vartheta_{ref} \in \{-0.1, 0, 0.1\}$rad. Since the reference value does not affect the dynamics of the system, we simply replicate the data previously collected for each reference value.

4.2 Training

Once the input data have been collected, we need to compute the output values. The first step is to consider the instantaneous rewards. The reward function is:

$$\mathscr{R}(s_t, a_t) = \begin{cases} -\left|\vartheta_t - \vartheta_{ref}\right| & \text{when } s_{t+1} \text{ not final} \\ -1 & \text{when } s_{t+1} \text{ final} \end{cases} \tag{6}$$

where a final state is the one in which the magnitude of angle ϑ exceeds 0.5 rad. As described in Section 2, fitted Q-iteration algorithms iteratively extend the time horizon by approximating Q-functions defined according to the following equation:

$$Q_k(s,a) = \mathscr{R}(s,a) + \gamma \cdot \max_b Q_{k-1}\left(s'_t, b\right). \tag{7}$$

On the basis of the approximation of the Q_{k-1}-function, it is possible to build the training set in order to get an approximation of the Q_k-function.

The first *Q-function*, Q_1, is the approximation of the direct rewards as calculated in (6). The training values of the following functions (Q_k) are determined by (7), using direct rewards and the approximation given in the previous step (Q_{k-1}). These values are the output values of Q_k used by the approximator.

For each experiment, we have approximated the Q-functions from Q_0 to Q_{50} using two kinds of function approximators: neural networks and extra-trees, which are briefly described in the following. For more details refer to [9, 2].

4.2.1 Training with neural networks

The training of neural networks follows the approach used for the *NFQ* algorithm [9]. The *Q-function* at the k^{th} step is represented by a neural network whose input is the tuple $(\dot{x}, \vartheta, \dot{\vartheta}, \vartheta_{ref}, C)$. The model considered for the network uses 2 hidden layers composed of 5 neurons each and an output layer composed of one neuron. The activation function is sigmoidal for the inner layers and linear for the output layer. The training method used to determine the set of weights and biases is Levenberg-Marquardt [7].

4.2.2 Training with Extra-Trees

Besides neural networks, we have performed experiments with extra-trees, a particular kind of regression tree ensemble. Each tree is built so that each test at a node is determined by selecting K candidate tests at random, and choosing the one with the highest score. The parameters used in our experiments are those proposed in [2]: 50 trees, 5 candidate tests, and each node must contain at least two samples.

5 Simulation Results

In this section, we present and discuss some of the results obtained with the fitted Q-iteration algorithm using neural networks (NN) and with extra-trees. To give an idea of the performances achievable by the learned controllers, in each graph we report three simulations, which correspond to controllers learned using datasets with different sizes. To compare the results, we show simulations starting from a fixed angular position: 0.2 rad.

Figure 2 compares the behavior of the LQR control with the behavior of the learned controllers when the angular set point is fixed to 0. It can be noticed that all the learned controllers are much faster than LQR to reach the set point. In particular, extra-trees get very close to the set point after only a few control steps, and neural networks take about one second to converge. On the other hand, using neural networks the controllers are much smoother than those achieved by extra-trees.

Figures 3 to 5 show the behavior with ϑ_{ref} varying according to different profiles. It is worth noting that all the controllers are able to approximately follow the given profiles, even if they have been trained only to follow three angular set points: $-0.1, 0, 0.1$. However, as we can see, neural networks are much more accurate (almost overlapping with the reference profile) than extra-trees. Extra-trees perform

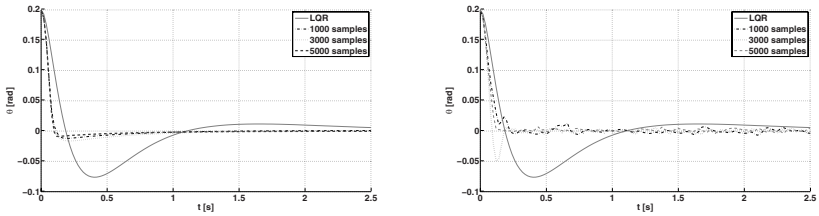


Fig. 2 Performance with $\vartheta_{ref} = 0$ (left:NN, right:Extra-Trees)

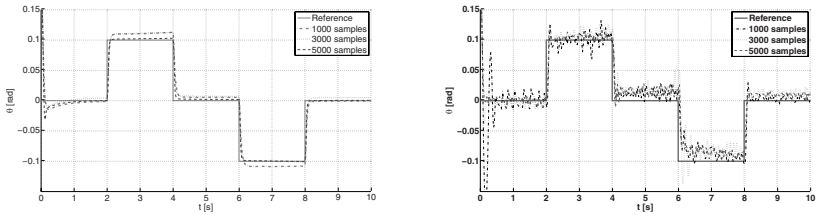


Fig. 3 Performance with ϑ_{ref} piecewise constant (left:NN, right:Extra-Trees)

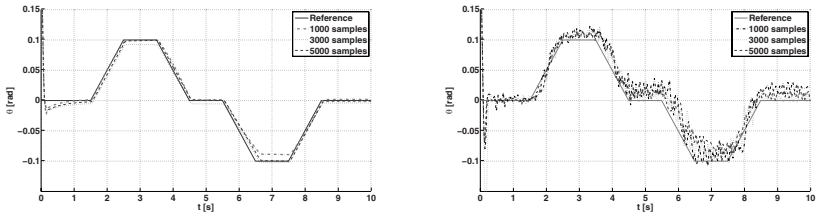


Fig. 4 Performance with ϑ_{ref} piecewise ramp (left:NN, right:Extra-Trees)

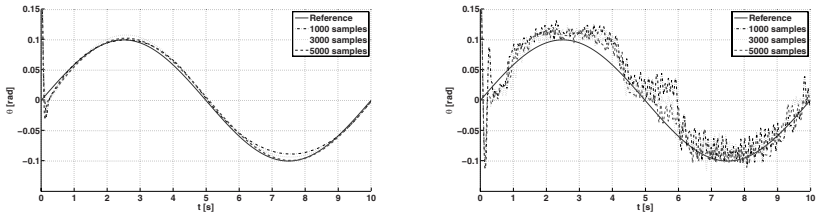


Fig. 5 Performance with ϑ_{ref} sinusoidal (left:NN, right:Extra-Trees)

quite poorly since they produce policies that make the robot reach speeds higher than those experienced using random exploration during the training phase, thus requiring hard extrapolation capabilities. This problem could be overcome by us-

ing the learned controller to collect and add further samples to the training set, and restarting the fitted Q-iteration algorithm.

As expected, controllers trained with larger datasets have better performances, even if it is worth noting that $1,000$ samples (corresponding to $20s$ of real time acquisition) are enough to learn quite good controllers.

6 Conclusions

In this paper, we presented batch RL methods to solve a robot control problem. The system considered here is unstable and non-linear, thus classic controllers require an approximated model. RL methods do not need any model of the robot and overcome problems of parameter identification. RL methods are generally used to solve single-task problems, while controllers generally follow changing reference points. We extended the idea of reference following to RL. The experiments show that a few tens of seconds are enough for batch RL algorithms to learn good controllers (even better than a classic controller like LQR). In particular, we have proposed a novel procedure that allows to learn controllers able to follow a varying reference point. It is interesting to note that the learned controllers effectively generalize to reference point not considered in the training phase. Given these encouraging results, we will experiment the proposed approach on the real robot.

References

1. Baird, L.C.: Residual algorithms: Reinforcement learning with function approximation. In: Proceedings of the 12th Intl. Conf. on Machine Learning, pp. 30–37 (1995)
2. Ernst, D., Geurts, P., Wehenkel, L.: Tree-based batch mode reinforcement learning. Journal of Machine Learning Research **6**, 503–556 (2005)
3. Gordon, G.J.: Approximate solutions to markov decision processes. Ph.D. thesis, Carnegie Mellon University (1999)
4. Landau, L., Lifshitz, E.M.: Mechanics, Course of Theoretical Physics, Volume 1. Pergamon Pres (1976)
5. Ogata, K.: Modern Control Engineering (4th ed.). Prentice Hall PTR, Upper Saddle River, NJ, USA (2001)
6. Ormoneit, D., Sen, S.: Kernel-based reinforcement learning. Machine Learning **49**(2-3), 161–178 (2002)
7. Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T.: Numerical Recipes: The Art of Scientific Computing. Cambridge Univ. Press, New York, 1989. (1989)
8. Reddy, J.: Energy Principles and Variational Methods in Applied Mechanics (2nd ed.). John Wiley and Sons, Hoboken, NJ, USA (2002)
9. Riedmiller, M.: Neural fitted q iteration - first experiences with a data efficient neural reinforcement learning method. In: Proceedings of European Conference on Machine Learning, pp. 317–328 (2005)
10. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA (1998)

KNOWLEDGE MANAGEMENT

Optimizing Relationships Information in Repertory Grids

Enrique Calot [1], Paola Britos [2], and Ramón García-Martínez [3]

Abstract The Repertory Grid method is widely used in knowledge engineering to infer functional relationships between constructs given by an expert. The method is ignoring information that could be used to infer more precise dependencies. This paper proposes an improvement to take advantage on the information that is being ignored in the current method. Furthermore, this improvement fixes several other limitations attached to the original method, such as election in a discrete set of two values as a similarity pole or a contrast pole, the arbitrary measurement of distances, the unit-scale dependency and the normalization, among others. The idea is to use linear regression to estimate the correlation between constructs and use the fitness error as a distance measure.

1 Introduction

The Repertory Grid method is widely used in knowledge engineering to infer functional relationships between constructs given by an expert. The the original method is ignoring information that could be used to infer more precise dependencies [1], [2] and [3].

This paper proposes an improved method using linear regression to calculate the dependencies using the given values and interpreting the scales and units. Vectorial constructs like colors and location are also be supported by this method.

[1] Enrique Calot

Intelligent Systems Laboratory. School of Engineering. University of Buenos Aires. ecalot@fi.uba.ar

[2] Paola Britos

Software & Knowledge Eng. Center. Buenos Aires Institute of Technology. pbritos@itba.edu.ar

[3] Ramon Garcia-Martinez

Software & Knowledge Engineering Center. Buenos Aires Institute of Technology. rgm@itba.edu.ar

Please use the following format when citing this chapter:

Calot, E., Britos, P. and García-Martínez, R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 163-172.

To do that the relationships are described using a relationship function or "model" with coefficients to bee calculated using the least squares method. The residuals that could not have been fit by the regression are the ones not explained by the model and are used to calculate a better measure of the distance. In Sec. 4 a use case is provided.

2 Deficiencies of the original repertory grid method

The following paragraphs enumerate some deficiencies in the original repertory grid method, the goal is to solve most of them.

2.1 Generated trees rely on the scale and units inherent to the data, which has been arbitrarily normalized

When comprising distances between the constructs c_1 and c_2 there could be errors resulting from the measurement using different magnitudes that make the data sensible to scale changes, in the repertory grid method this issue is ignored simply taking the numerical values without checking the unit; even though this could be dangerous because the results may be different depending on what units the expert is measuring on, even when normalized to numbers from 1 to 5. e.g. Celsius, Fahrenheit and Kelvin scales measure temperature, but the zeros are in different places, so the measured values are not proportional. Even when normalized, the values will be different depending on the expert's scale choice. Another examples are the logarithmic scales such as pH, decibels and even musical notes.

2.2 Vectorial constructs are not supported

There are constructs that have a vectorial nature such as colors, coordinates and so forth. They cannot be reduced to a linear scale because they could not only depend on the way they were converted to a scalar value but also much information could be lost (and degrees of freedom). e.g. a color may be represented in values like 1=red 3=blue 5=black, but the sense of that measure is completely lost. Neither can be separated into different components and be studied independently because one of the components is inherently related to the others and the construct must be studied as a whole and not by the parts. e.g. A dead pixel may depend on the intensity of the red value (in its RGB scale) but the human eye percepts the colors better using the HSL scale. When measuring in HSL it is very unlikely to see the direct correspondence between the probability to lose a pixel and the red component of the color if it is measured in the HSL scale. Studying the HSL as a whole should find an association and then will be possible to realize that this

relation is very similar to the red transformation function in the HSL to RGB conversion method.

2.3 It is discrete

The grid could be much better generalized if continuous values are used. Greater precision could be acquired when comparing our results.

2.4 The distance measurement is a kind of dubious

Using the 1-norm is arbitrary. How do we know that this is the better choice?

3 The Proposed Method

The objective of the repertory grid method is to find functional relationships between constructs, the original method proposes the equality between two constructs as the optimal dependency and then measures how deviated are the constructs to each other using the 1-norm.

This paper, in contrast, proposes the use of a regression method to fit the given data using the resultant fitness as a measure of the relationship between the adjusted constructs.

3.1 Definitions on matrices

Before stating the method some definitions on the original repertory grid method are to be explained.

3.1.1 Repertory grid matrix

Let G be the grid matrix. It has n elements and m characteristics. The notation g_{G_i,C_j} with $0 \le i < n, 0 \le j < m$ to each of its elements will be used.

3.1.2 Distance matrix

Let D be the matrix containing the distance between characteristics. It has m columns and m rows, one for each characteristic. It is superior triangular without diagonal, so d_{C_i,C_j} with $0 \le i < m, i < j < m$ for each of its elements. In the original repertory grid method it is the 1-norm distance between two columns (i and j) in the G matrix.

When adding the trivial twist to support the contrast construct, it should use the minimum value between the 1-norm distance from the first construct to both: the second construct and its contrast (each value of this column C_i is replaced by $6 - C_i$).

3.2 Measuring distances

The measurement is based on the hypothesis that F(Ci,Cj)=1 where Ci and ,Cj are two constructs, then it measures how deviated the is fitness to the hypothetical value (1 in this case). The obtained residue should reflect how both constructs are explained by a model and the degree of dependency between both constructs.

Before doing the measurements, the knowledge engineer should define a model, that is equivalent to state arbitrarily the equations where the fitting will be made. Defining a model is the most important step in the method, because the measurements not only depend on the relationships between the constructs but on how the model applies to the situation. It is possible to use different models to measure distances between different pairs of constructs; the method should support such case.

The distance matrix could be filled by the fitness error. This is really an excellent way to measure the dependency between two characteristics. If the model were completely generic (ideal, but impossible), the fitness error would be the optimal distance to measure the dependence. This is an ideal case and not useful in practice, it is recommended to use simple models to find real relationships and avoid complex equations.

3.3 Defining a fitting model

To calculate the distance between two constructs the knowledge engineer must define a model. In the original method, the model was linear, scalar and discrete. Let $\zeta_{U,V}$ be the model to correlate variables U and V. Examples of models are the linear $\zeta_{U,V}=\{1,u,v\}$ and the quadratic $\zeta_{U,V}=\{1,u,u^2,v,v^2,uv\}$.

The cardinality n of $\zeta_{U,V}$ is the number of coefficients to be calculated by our regression. Let $\vec{\zeta}_{U,V}\in\Re^n$ be the vector representing that model and $\alpha_{\zeta_{U,V}}\in\Re^n$ the vector of coefficients represented by that model. These definitions arrive to the current ideal equation

$$F(C_1,C_2)=1=\alpha_{\zeta_{C_1,C_2}}\vec{\zeta}_{C_1,C_2} \quad (1)$$

if there is a $\alpha_{\zeta_{C_1,C_2}}$ that satisfies the equation for all pair of c_1 and c_2, then the fitness is perfect.

For example, the linear model $\zeta_{U,V}=\{1,u,v\}$ derives to the plane

$$F(U,V) = 1 = \alpha_1 + \alpha_2 u + \alpha_3 v \quad (2)$$

and the quadratic $\zeta_{U,V} = \{1, u, u^2, v, v^2, uv\}$ to

$$F(U,V) = 1 = \alpha_1 + \alpha_2 u + \alpha_3 u^2 + \alpha_4 v + \alpha_5 v^2 + \alpha_6 uv \quad (3)$$

Let ζ_W be the one-variable model related to the model $\zeta_{U,V}$ and may be obtained by

$$\zeta_W = \zeta_{U,V} \lfloor_{U=W,V=0} \cup \zeta_{U,V} \lfloor_{U=0,V=W} \quad (4)$$

that is $\zeta_W = \{1, w\}$ for the linear model and $\zeta_W = \{1, w, w^2\}$ for the quadratic.

3.4 Limitations of the proposed method

The method is linear since linear regression has been used. This means that the resultant relationships will be shown in euclidean subspaces resulting from the sum of terms with the form of coefficient α_n multiplied by a function dependent of the input data. This function is part of the model and does not necessarily need to be linear. It is not on the scope of this paper to study non-linear dependencies.

3.5 Calculating the regression

The least squares method should find the best fit. The matrix A related to two constructs is calculated by the evaluation of each construct's value in the desired model. The matrix is calculated column by column for each row in G as

$$A_i = \vec{\zeta}_{U,V} \lfloor_{U=G_{C_1,i}, V=G_{C_2,i}} \quad (5)$$

which is exactly evaluating the model (except the first 1) with the values of each row from the repertory grid matrix.

Finally the coefficients may be obtained by the multiplication of the pseudo-inverse matrix [2] and the unit vector.

$$\alpha = (A^T A)^{-1} A^T \vec{1} \quad (6)$$

The first value in the model (the constant part) must not be used because it will be in the other side of que equation as the unit vector $\vec{1}$. By doing that resultant equation Eq. (1) representing the model has been calculated.

3.6 Measuring the residuals

The desired measure of the fitness may be expressed by the residuals, that is the difference between the model evaluated with the repertory grid elements and the ideal result that is the unit vector.

$$\alpha \vec{\zeta}_{U,V} = \vec{1} + \vec{\varepsilon}_{U,V} \Rightarrow \mathrm{P}\alpha\vec{\zeta}_{U,V} - \vec{1}\mathrm{P} = R_{U,V} \quad (7)$$

where $R_{U,V} = \mathrm{P}\vec{\varepsilon}_{U,V}\mathrm{P}$.

The first impression is that $R_{U,V}$ is a good measure of the correlation between U and V but the fact is that it is a good measure of "what may not be explained by the model". It is possible that a construct is very attached to itself, its variance is very small and therefore the fitness is very small too.

For example, in the linear model $\zeta_{U,V} = 1, u, v$, it is possible that the construct U has values strictly around 6 for all of its elements. The resultant plane will be $\frac{u}{6} + 0v = 1$. In this case $R_{U,V}$ represents the fitness of U by itself and not the fitness of U related to V.

This paper proposes the use of two more regressions, the one related to U and the one related to V with the residuals R_U and R_V respectively. Having known that regressions, a good redefinition of the distance could be "what is explained by the two-variable model that was not already explained by each separated variable using a single variable model". This is the definition

$$d_{i,j} = \frac{\min\{R_{C_i}, R_{C_j}\}}{R_{C_i,C_j}} \tag{8}$$

with $R_{C_i} > R_{C_i,C_j}$ and $R_{C_j} > R_{C_i,C_j}$ because the least squares method had minimized R_{C_i,C_j} with more degrees of freedom. It is easy to show that the distance matrix will have values between 0 and 1, being 0 the stronger relationship according to the chosen model and 1 the weakest one.

3.7 Vectorial constructs

In vectorial constructs the process is mostly the same, the only difference is that the whole vector must be evaluated in the model for each component. That is for example a construct color $\vec{C} = (R, G, B)$ and another scalar lightness L should be evaluated in the quadratic model as

$$\alpha_1 r + \alpha_2 r^2 + \alpha_3 g + \alpha_4 g^2 + \alpha_5 b + \alpha_6 b^2 + \alpha_7 l + \alpha_8 l^2 = 1.$$

The residuals should be divided by the minimum of the regression of each separated construct, in the example by the residual of

$$\alpha_1 r + \alpha_2 r^2 + \alpha_3 g + \alpha_4 g^2 + \alpha_5 b + \alpha_6 b^2 = 1 \text{ and } \alpha_1 l + \alpha_2 l^2 = 1.$$

3.8 Normalization is no longer needed

A side effect by the use of a dependency function is that the units and scales are inside the model being completely abstracted to the method. There is no longer need to have a discretized input of numbers from 1 to 5, now, the normalization is

inside the method which will find the best fit regardless the scale and the units. If the scale is logarithmic adding a logarithm to the model should be enough.

4 Case Study

Our expert is providing the knowledge engineer with four constructs; the first one is the vectorial location of a city ($\vec{L} \in \Re^2$), its population ($P \in N$), temperature ($T \in \Re$) and the average level of pollution ($O \in \Re$). The obtained values are shown in Table 1.

Table 1. Population, temperature and pollution of a city regarding to its location on an arbitrary coordinate plane.

Location km;km	**Population** Hab	**Temperature** $^\circ$C	**Pollution** ppm
(9.83982;40.4372)	73272	30.8322	36.8086
(17.3862;69.5633)	65115	27.916	41.1447
(24.1684;89.5489)	94737	25.7233	42.2755
(26.9449;57.6548)	85173	25.5949	28.7814
(47.1808;33.3024)	102663	29.8456	15.5273
(67.8653;72.7391)	118860	27.717	24.7249
(48.1759;80.8657)	19293	24.4881	26.9948
(16.3168;20.8034)	105084	29.317	28.6556
(28.1486;43.4684)	57170	29.8976	29.4861
(55.1659;3.49954)	3431	30.2999	17.1146
(45.7604;63.8792)	4965	26.7262	25.5383

The knowledge engineer calculates the regressions and compares the results as shown in Table 2. Finally, as the temperature quadratic model has a very small variance by itself, the engineer decides to use the linear model for this construct. To calculate the distances between constructs the smallest one-construct residual is taken to divide the two-construct to be measured. The resultant distance matrix calculated by all this divisions is shown in Table 3.

Finally we perform the tree building method as shown in Fig. 1. As we can see, the Pollution is primarily related to the location, then to the temperature and finally to the population. In Fig. 2 it is shown the level of pollution over a region of $\vec{L}$, deduced from the resultant subspace, in Fig. 3 the temperature is shown under the linear model and in Fig. 4 under the wrong quadratic model which had been discarded by the knowledge engineer.

The pollution equation suggests that the proximity to (60km;20km) has low pollution, perhaps it is the top of a mountain. As we can observe, the method found the dependencies.

Table 2. Comparison between models.

Construct	Model	**Residual**	**Subspace**
T	Quadratic	0.0144453	$0.0723193\ t - 0.00130023\ t^2 = 1$
T	Linear	0.244034	$0.035479\ t = 1$
P	Quadratic	1.37606	$0.000030232\ p - 1.9863\ 10^{-10}\ p^2 = 1$
O	Quadratic	0.291629	$0.0697022\ c - 0.00113199\ c^2 = 1$
$\vec{L}_1$	Quadratic	0.733654	$0.0574155\ l_1 - 0.000689149\ l_1^2 = 1$
$\vec{L}_2$	Quadratic	0.979385	$0.0391949\ l_2 - 0.00033698\ l_2^2 = 1$
$\vec{L}$	Quadratic	0.560544	$0.0356493\ l_1 - 0.000410520\ l_1^2 + 0.0168284\ l_2 - 0.00015598\ l_2^2 = 1$
$\vec{L},T$	Quadratic	0.0137924	$0.0722\ t - 0.0013\ t^2 + 0.0002\ l_1 - 3.0117\ 10^{-6}\ l_1^2 - 0.0001\ l_2 + 1.403\ 10^{-6}\ l_2^2 = 1$
$\vec{L},P$	Quadratic	0.43544	$0.000013\ p - 9.1028\ 10^{-11}\ p^2 + 0.0308\ l_1 - 0{,}0003\ l_1^2 + 0.0033\ l_2 - 0.00035\ l_2^2 = 1$
$\vec{L},O$	Quadratic	0.1191	$0.03239\ o - 0.0002\ o^2 + 0.0176\ l_1 - 0.0001\ l_1^2 + 0.0008\ l_2 - 0.00004\ l_2^2 = 1$
T,O	Quadratic	0.0143339	$0.00039\ o - 7.008\ 10^{-6}\ o^2 + 0.0719\ t - 0.0013\ t^2 = 1$
P,O	Quadratic	0.286957	$0.0687\ o - 0.00111\ o^2 - 4.633\ 10^{-7}\ p + 6.992\ 10^{-12}\ p = 1$
T,P	Quadratic	0.0128094	$8.9589\ 10^{-8}\ p - 1.1170\ 10^{-12}\ p^2 + 0.0723\ t - 0.0013\ t^2 = 1$
$\vec{L},T$	Linear	0.12522	$0.0309\ t + 0.0006\ l_1 + 0.002\ l_2 = 1$
$\vec{L},T$	Combined	0.0952247	$0.0291\ t + 0.0063\ l_1 - 0.00008\ l_1^2 + 0.0009\ l_2 + 0.00001\ l_2^2 = 1$
O,P	Combined	0.155222	$0.0289\ o - 0.0005\ o^2 + 0.0204\ t = 1$
T,P	Combined	0.243501	$4.78\ 10^{-7}\ p - 3.855\ 10^{-12}\ p^2 + 0.0351\ t = 1$

Table 3. Distance as relationships between constructs using an arbitrary model.

	$\vec{L}$	P	T	O
$\vec{L}$		0.776819	0.390211	0.212472
P			0.997819	0.98398
T				0.636069
O				

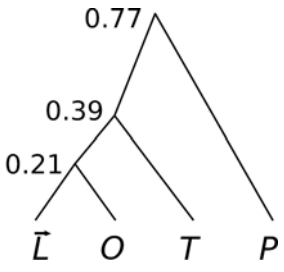


Fig. 1. Tree view of the distances built by the proposed repertory grid method.

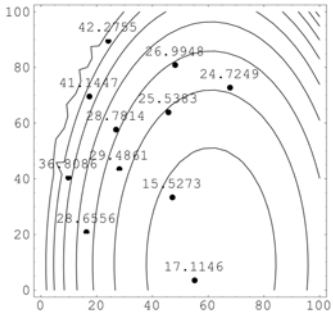


Fig. 2. Regressed pollution depending on the location.

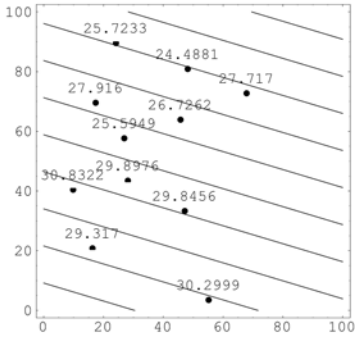


Fig. 3. Regressed temperature under a linear model depending on the location.

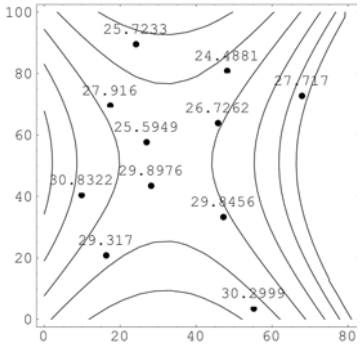


Fig. 4. Regressed temperature under a wrong quadratic model depending on the location.

5 Conclusions

The proposed method has potential application on several fields, specially in knowledge acquisition. The usage of pre-designed model instead of the discrete-linear one may fit with more constructs and helps the knowledge engineer in the exploration of the construct. Future lines of development may find better ways to choose the appropriate model. Using the fitness coefficient as a measure is refined generalization of the method.

References

1. Bradshaw, J.M., Ford, K.M., Adams-Webber, J.R. and Boose, J.H. *Beyond the repertory grid: new approaches to constructivist knowledge acquisition tool development*. International Journal of Intelligent Systems 8(2) 287-33. (1993).
2. Acton. F. S. *Analysis of Straight-Line Data*. Dover Publications. (1966).
3. Beeri, C., Fagin, R. and Howard J. H. (1977). *A complete axiomatization for functional and multivalued dependencies in database relations*, In Proceedings of the 1977 ACM SIGMOD international Conference on Management of Data (Toronto, Ontario, Canada, August 03 - 05, SIGMOD '77. ACM, New York, NY, 47-61. (1977).
4. Barlett. D. *General Principles of the Method of Least Squares.* Dover Publications. (2006).

Modeling Stories in the Knowledge Management Context to Improve Learning Within Organizations

Stefania Bandini, Federica Petraglia, and Fabio Sartori

Abstract Knowledge Management has been always considered as a problem of acquiring, representing and using information and knowledge about problem solving methods. Anyway, the complexity reached by organizations over the last years has deeply changed the role of Knowledge Management. Today, it is not possible to take care of knowledge involved in decision making processes without taking care of social context where it is produced. This point has direct implications on learning processes and education of newcomers: a decision making process to solve a problem is composed by not only a sequence of actions (i.e. the know-how aspect of knowledge), but also a number of social interconnections between people involved in their implementation (i.e. the social nature of knowledge). Thus, Knowledge Management should provide organizations with new tools to consider both these aspects in the development of systems to support newcomers in their learning process about their new jobs. This paper investigates how this is possible through the integration of storytelling and case-based reasoning methodologies.

1 Introduction

Storytelling is a short narration through which an individual describes an experience on a specific theme. In this way, the human being is motivated to focus the attention on his/her own knowledge about the specific theme that is the subject of narration [5]. Within organizations, storytelling can be considered an effective way to treasure the knowledge that is produced from the daily working activities. For

Stefania Bandini
CSAI, Viale Sarca 336, 20126 Milan (ITALY), e-mail: bandini@csai.disco.unimib.it

Federica Petraglia
DISCO, Viale Sarca 336, 20126 Milan (ITALY), e-mail: federica.petraglia@csai.disco.unimib.it

Fabio Sartori
DISCO, Viale Sarca 336, 20126 Milan (ITALY), e-mail: sartori@disco.unimib.it

Please use the following format when citing this chapter:

Bandini, S., Petraglia, F. and Sartori, F., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 173–182.

example, Roth and Kleiner [9] have analyzed how the adoption of storytelling allows an organization to be more conscious about its overall knowledge, to share knowledge among all the people involved in its generation, to treasure and disseminate new knowledge originated by the sharing of different stories. The adoption of storytelling can promote the development of new professional contexts where different professionals collaborate to solve common problems, share experiences, explicit and implicit assumptions and understandings in order to improve the global capability of the organization to transform, create and distribute knowledge. In this sense, Knowledge Management can profitably exploit the storytelling as a way to make explicit the individual experiences, skills and competencies, promote the negotiation processes through dialogues among people involved, support the reification of new knowledge in order to make it available for the future and help newcomers in the learning process about his/her job through the analysis of the problem–solving strategies and social context represented by the stories. In this paper, we present a conceptual and computational framework for supporting continuous training within wide organizations, in the learning by doing [12] context. This approach is based on the integration of storytelling and case–based reasoning [10] methodologies: the former allows to manage a decision making process like a story that describes problem characteristics and what kind of communications among people and problem solution strategies can be applied to solve it; the latter is a very useful and efficient mean to compare stories (i.e. cases) finding solutions to new problems by reusing past experiences. Next section is devoted to make clear how learning by doing, storytelling and case based reasoning can be put together; first, a brief introduction to learning by doing and historical/methodological motivations to adopt it as a good paradigm for supporting continuous learning in organization is given. Then, its relationship with storytelling and case based reasoning is explored in details, to show how storytelling is the theoretical bridge between the need to support learning by doing through computer-based tools and one of the most suitable computer science paradigm for this scope. In section 3, an application of the framework to the SMMART (System for Mobile Maintenance Accessible in Real Time) project will be briefly introduced, to show its effectiveness in representing problem solving strategies of experts in the form of stories that can be archived as cases into a case base and used as pieces of experience to build newcomers training systems, according to the learning by doing approach. In particular, the domain of the SMMART project is the troubleshooting of trucks (thanks to the collaboration with Volvo Trucks), thus the stories involved concern the experience owned by expert mechanics and the system is devoted to support newcomers of a truck manufacturers after-sales department. Finally, conclusions and future work will be briefly pointed out.

2 Learning by Doing, Storytelling and Case Based Reasoning

Contemporary socio-cultural context supports the idea of knowledge acquisition and management, not only as development of organisation, policy, methods of knowl-

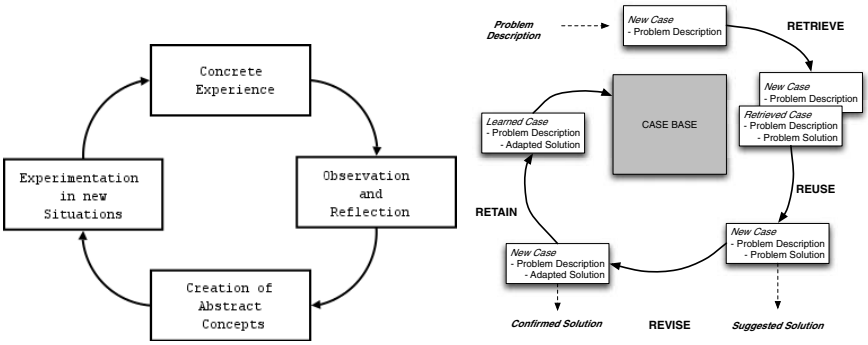


Fig. 1 On the left, the four steps in learning by doing methodology; on the right, the 4R's cycle of CBR applications

edge diffusion, but also as a communitys benefit. Starting from these considerations, we reflect about the concept of continuous learning within organizations and how to support it. In particular, we focus the attention on learning by doing paradigm. Learning by Doing is based on well known psycho–pedagogical theories, like cognitivism and behaviourism, which are devoted to point out the role of practice in humans' intellectual growth and knowledge improvement. In particular, this kind of learning methodology refuses the typical idea that concepts are more fundamental than experience and, consequently, that only a solid set of theoretical notions allows to accomplish a given task in a complete and correct way. Learning by doing methodology states that the learning process is the result of a continuous interaction between theory and practice, between experimental periods and theoretical elaboration moments. Learning by doing can be articulated into four distinct steps (see the left part of Figure 1), where practical phases (i.e. Concrete Experience and Experimentation) are alternated with theoretical ones (i.e. Observation and Reflection and Creation of Abstract Concepts): starting from some kind of experience, this experience originates a mind activity that aims to understand the phenomenon; this step ends when a relation between the experience and its results (typically a cause-effect relation) is discovered that can be generalized to a category of experiences similar to the observed phenomenon. The result is a learned lesson that is applicable to new situations which will eventually occur in the future.

In our framework, a concrete experience can be represented by a story, which represents a decision making process about a problem to be solved. This story should give to a newcomer an idea of how a critical situation could be tackled, according to the knowledge owned by experts. Moreover, it could give indications about who could help him/her in case of need.

Stories can be archived as cases according to the case-based reasoning (CBR) paradigm. Case Based Reasoning is an Artificial Intelligence method to design knowledge management systems, which is based on the principle that *similar problems have similar solutions*. For this reason, a case based system doesnt require a

complete and consistent knowledge model to work, since its effectiveness in finding a good problem solving strategy depends typically on how a problem is described. Thus, CBR is particularly suitable to adopt when domains to tackle are characterized by episodic knowledge and it has been widely used in the past to build decision support systems in domain like finance [4], weather forecasting [8], traffic control [7], chemical product design and manufacturing [3], and so on. A case, is a complete representation of a complex problem and it is generally made of three components: description, solution and outcome [10]).

The main aim of CBR is finding solutions to new problems through the comparison of it with similar problems solved in the past, as shown in the right part of Figure 1, that is the well known 4Rs cycle by Aamodt and Plaza [1]: the comparison is made according to a *retrieval algorithm* working on problem features specified in the description component. When an old problem similar to the current one is retrieved, its solution is reused as a solving method for the new problem. The solution can be then *revised* in order to fit completely the new problem description and finally *retained* in the case base to become a sort of new lesson learned. In the retained case, the outcome component gives an evaluation about the effectiveness of the proposed solution in solving the problem. In this way, new cases (i.e. stories) can be continuously created and stored to be used in the future, building up a memory of all experiences that can be used as newcomer training tool.

Starting from concrete experiences newcomers can learn decision making processes adopted within the organization they are introducing quicker than studying manuals or attending courses. Moreover, the comparison between their own problem solving strategy and the organization one, represented by the collection of stories, stimulates the generalization of problems and consequently the reflection about general problem solving methods, possibly reducing the time period to make the newcomers able to find effective solutions.

CBR is one of the most suitable Artificial Intelligence methods to deal with learning by doing [11], due to the perfect match between their cycles of life. In particular: the description of a new case can be a way to represent experimentation in new situations, since the aim of CBR is to solve a new problem exploiting old solutions to similar problems. Thus, a new case is the attempt to apply past experiences to a new concrete situation in order to validate a problem solving strategy, as the experimentation in new situations is a way in the learning by doing context to test the generation of abstract concepts starting from already validated concrete experiences; a retrieved case in the case base represents a concrete experience in the learning by doing framework; retrieval, reuse and revise are the CBR phases during which a solution to a new problem is found and reused by comparison with similar past problems and then adapted to fit completely the critical situation defined by problem description. Thus, they can be exploited to model the theoretical steps of learning by doing methodology (i.e. Observation/Reflection and Creation of abstract concepts), through which a newcomer finds a general way to tackle a problem starting from a set of existing examples; finally, the retained case in the CBR paradigm is the completion of the initial problem to be solved with the optimal solution ob-

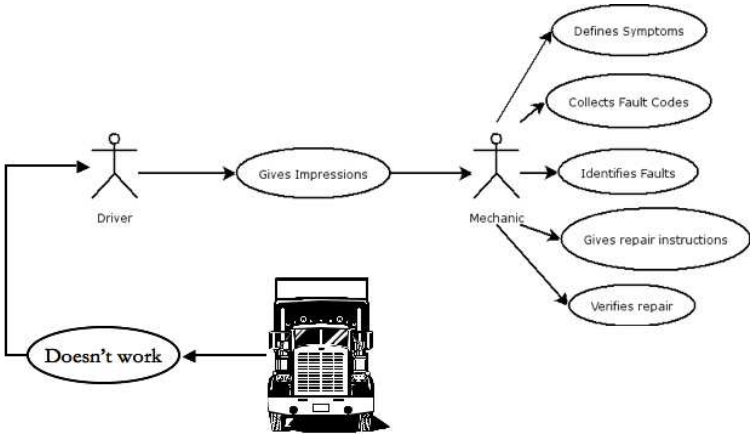


Fig. 2 A typical story about a truck troubleshooting session

tained at the end of the CBR cycle, thus it represents a new instance of the initial experimentation in new situations.

Moreover, since the concept of story can be used to describe both a case in the CBR paradigm and a concrete experience in the learning by doing methodology, in our opinion, storytelling is the optimal connection between a case-based support to the development of training systems for newcomers and the learning by doing context.

3 The SMMART Project

SMMART (System for Mobile Maintenance Accessible in Real Time) is a research project funded by the European Community[1] that aims to develop a decision support system for supporting experts of Volvo Truck[2] , a world leader in the manufacturing of trucks in troubleshooting vehicle problems. To this aim, a case-based reasoning module of the final system is going to be designed and implemented in order to detect the most probable faulty engine component on the basis of a given set of information, which can be archived as a story.

The narration (see Figure 2) about the problem starts when a driver recognizes that a problem arose on his/her truck truck. For example, a light of the control panel turns on or some unpredictable event happens (e.g. smoke from the engine, oil loss, and noises during a break and so on). Thus, the driver contacts the truck after sale assistance to obtain problem solution. The mechanic who receives the truck is responsible for making a detailed analysis of it by taking care of driver impressions,

[1] Project number NMP2-CT-2005-016726

[2] http://www.volvo.com/trucks/global/en-gb/

testing it and collecting information coming from on–board computers. Then, he/she has to find the fault, repair it and verify that the problem has been solved before the truck leaves the workshop. In the following, a detailed description of how such stories have been represented and used in the context of SMMART is given in terms of case structure and similarity functions developed.

3.1 The Case Structure: a story in the SMMART Context

The final scope of the CBR tool is to identify the most probable truck faulty component (e.g. engine, gearbox), namely *High Level Component* (HLC). The HLC is an indication where the real cause of the truck malfunction is: this is the *root cause* and it is detected by the mechanic manually or through the exploitation of traditional softwares used by Volvo repair shops. Anyway, the CBR systems archives all the information about the problem, in order to give a complete representation of the story involved, as shown in Figure 3: HLC and root cause represent the *solution* part of the case, while the problem analysis made by mechanic, that is represented as the *case description*, considers four main categories of information: *symptoms*, *fault codes*, *general context* and *vehicle data*.

Symptoms give qualitative descriptions of truck problems and their context. For example, the sentence "The truck cruise control fails to maintain set speed while driving uphill at -20C under heavy charge" specifies that a possible fault of the cruise control (i.e. the symptom) is detected when the road is not plane, the temperature is very low, and the truck is transporting a big load (i.e. the context). The same problem could be not detected under different conditions. Symptoms a grouped into a tree structure within the SMMART case description: currently, five levels are considered, but they could increase in the future.

Fault codes are quantitative information coming from on–board computers: when some event happens that possibly causes malfunctions, a fault code is generated and memorized to be used during troubleshooting sessions. A fault code is characterized by many fields, the most important are (1) the *Message IDentifier* (MID), that specifies the on–board computer generating the error code (for example, the entries with MID 128 in Figure 3 identify the on–board computer monitoring the engine; for this reason, it can be deduced that the MID indirectly identifies a HLC), (2) the *Parameter Identifier* (PID), that specifies which component of the on–board computer has generated the Fault Code (this means that an on–board computer is characterized by the presence of many sensors, each of them devoted to monitor a specific part of the HLC under control), and (3) the *Failure Mode Identifier* (FMI) that identifies the category of the fault (electrical, mechanical, and so on). The main activity of the mechanic during the truck analysis is the correlation between symptoms and their fault codes: in this way, it is possible to identify the faulty component, to repair it and trying to verify if the problem has been solved by controlling if fault codes disappear when the truck is turned on.

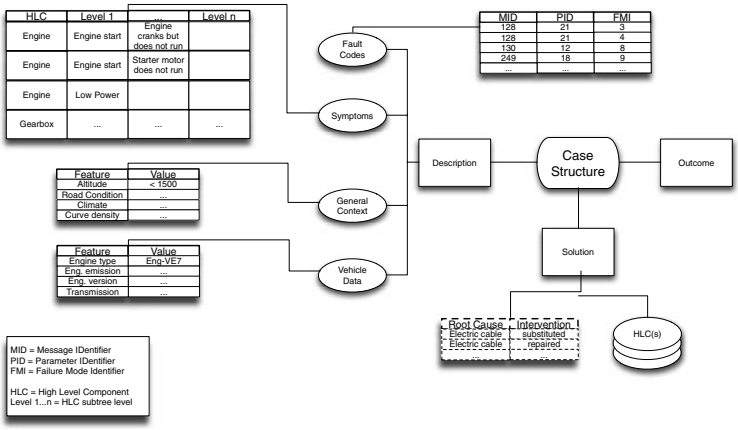


Fig. 3 The case structure of the SMMART project

Finally, general context and vehicle data contain information about driving conditions and truck characteristics respectively. These two kinds of information are not directly related to the fault generation, but they can be useful during the similarity calculus. For this reason, they have been included in the case description.

3.2 The Similarity Function: Retrieving Stories in the SMMART Context

When a new story is generated that represents the current problem (i.e. a problem without solution), it is represented as a case and properly described in terms of symptoms, fault codes and context. Then, it is compared with other cases already solved in the past in order to find similar story descriptions: the solution of most similar story is then reused as a starting point for deriving the solution to the current problem, suggesting in this way how to detect the most probable root cause. The comparison between stories is done according to a retrieval algorithm based on the K–Nearest Neighbor approach [6].

Given the current case C_c, for which no solution is given, the goal of the retrieval algorithm is to propose a possible solution (i.e. a HLC together with a possible root cause) by comparing its description C_c^d with the descriptions C_p^d of each case C_p solved in the past and included in the case base. The similarity among cases is calculated with a composition of sub functions, as described by the following formula

$$SIM(C_c, C_p) = \frac{k_1 * SIM_S + k_2 * SIM_{FC} + k_3 * SIM_{Vehicle} + k_4 * SIM_{GenContext}}{4}$$

where:

- $k_1 ... k_4$ are configurable weights and $k_1 + k_2 + k_3 + k_4 = 1$;
- SIM_S, SIM_{FC}, $SIM_{Vehicle}$ and $SIM_{GenContex}$ are in [0.0 ... 1.0].

SIM_S is the similarity among the two sets of symptoms of current case and past case, named S_c and S_p respectively: for each symptom A in the current case, the algorithm finds the closest symptom B (possibly the same as symptom A) in the past case, belonging to the same sub–tree, having the HLC name as its root.

The function *dist(A,B)* gives the minimum number of arcs that separates A and B in the symptoms tree and it is used for calculating the similarity. Similarity between symptom A and symptom B ($A \in S_c$ and $B \in S_p$) is

$$sim(A,B) = (1 - dist(A,B)/dmax)$$

where dmax is the constant maximum distance possible between two nodes in the tree (in the current symptom tree dmax=5). Similarity between symptom A and symptom B is modified by the conditions under which the symptoms occurred; the algorithm evaluates the degree of similarity between the two sets of conditions and modifies the value of sim(A,B) consequently.

The similarity among symptoms SIM_S is the sum of all the sim(A,B) normalized with the number *noc* of couples of symptoms considered and eventually penalized if the two cases are different in number of symptoms. The final formula is:

$$SIM_S = (SIM_S/noc) * (1 - Penalty)$$

where $Penalty = \frac{(\#S_c + \#S_p - 2*noc)}{\#S_c + \#S_p}$

SIM_{FC} is the similarity among the two sets of fault codes (FCs) calculated on each HCL group of FCs (FCs grouped by high level component): the relation between FCs and HLCs is given by mapping the MID of each FC to the HLC name. Doing so, different MIDs (that means FCs coming from different processing units) can be associated to the same HLC. If a FC has not any MID–HLC mapping entry, the FC will be related to a fictitious HLC, called HLC_0: in this way, also Fault Codes which cannot be linked directly to a specific HLC can be compared, with benefits from the final similarity point of view.

When all the Fault Codes of both C_c and C_p have been grouped in the FC_c and FC_p sets respectively, the algorithm compares the information they contains: the similarity sim(A, B) between two Fault Codes belonging to C_c and C_p depends on their PID and FMI values. The similarity values are fixed and they have been determined with the collaboration of Volvo Truck experts. The similarity among fault codes SIM_{FC} is the sum of all the sim(A, B) normalized with the number *noc* of couples of fault codes considered and eventually penalized if the two cases are different in the number of fault codes; The final formula is:

$$SIM_{FC} = (SIM_{FC}/noc) * (1 - Penalty)$$

where $Penalty = \frac{(\#FC_c + \#FC_p - 2*noc)}{\#FC_c + \#FC_p}$.

$SIM_{Vehicle}$ is the similarity among the two vehicle characteristics: each possible feature involved in vehicle description is linked to a weight. These weights are used in the computation of the similarity between vehicle descriptions given in the current case and in the past case.

$SIM_{GenContext}$ is the similarity among the two general contexts. Since items describing general contexts are assigned qualitative values (i.e. strings), these values are preprocessed according to an opportune mapping function to be converted an integer values.

4 Conclusions

This paper has presented a framework to support learning by doing within organizations; this framework is based on the integration of storytelling and case based reasoning methodologies. Storytelling has been chosen due to its capability of taking care of different kinds of knowledge in the description of working experiences and presenting important pieces of expertise to newcomers in wide organizations; according to Atkinson [2]:

> Storytelling is a fundamental form of human communication [...] We often think in story form, speak in story form, and bring meaning to our lives through story. Storytelling, in most common everyday form, is giving a narrative account of an event, an experience, or any other happening [...] It is this basic knowledge of an event that allows and inspires us to tell about it. What generally happens when we tell a story from our life is that we increase our working knowledge of ourselves because we discover deeper meaning in our lives through the process of reflecting and putting the events, experience, and feelings that we have lived into oral expression.

On the other hand, case based reasoning is one of the most suitable Artificial Intelligence paradigms to deal with episodic and heterogeneous knowledge and consequently, in our opinion, it is probably the best approach to manage unstructured narrations about expertise and problem solving strategies. The proposed framework provides newcomers with a complete representation of the competencies developed by experts over the years. Thus, they can increase their experience about the problem solving strategy used inside the organization as well as the understanding about who are the people to contact in case of need (i.e. the experts who solved similar problem in the past).

In order to test the effectiveness of our approach, its application in the context of the SMMART project has been briefly introduced. It is important to highlight that the SMMART projects aims at the development of a CBR module to identify the most probable faulting component of a truck by means of a specific retrieval algoeirthm: the solution of the CBR engine is not subject to adaptation, since it is not the real solution of the mechanic troubleshooting session. A mechanic exploits this solution as a starting point to make deeper analysis looking for the root cause. Anyway, once the mechanic detect the real cause(s) of the problem, the CBR module retains it in the case base together with all other information, in order to give

a complete representation of the story related to that troubleshooting session. From the learning by doing point of view, the case base composed of all the stories about past troubleshooting sessions is a very important source of knowledge for newcomers; they could be solicited to solve a problem by specifying what are the symptoms and the related fault codes. Then they could try to identify faulty components and then compare their solution with the one proposed by the system, with an immediate evaluation of their own capability to learn expert mechanics decision making processes and identification of points they have to work on, maybe asking directly to the people who solved past problems. In this way, experience and knowledge created by the organization over the years and captured by the CBR system could be used as a very important training method alternative to the more traditional ones.

Future works are devoted to verify the applicability of the proposed methodology in building supporting systems for learning by doing in other complex contexts.

References

1. Aamodt, A. and Plaza, E.: Case–Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications, Vol. 7, No. 1, pp. 39-59, (1994).
2. Atkinson, R.: The Life Story Interview, Sage University Papers Series on Qualitative Research Methods, vol. 44, SAGE Publications, Thousand Oaks, CA, (1998).
3. Bandini, S., Colombo, E., Sartori, F., Vizzari, G.: Case Based Reasoning and Production Process Design: the Case of P-Truck Curing, In: ECCBR Proceedings. Volume 3155 of Lecture Notes in Computer Science, pp. 504517. Springer, Heidelberg (2004).
4. Bonissone, P. P., Cheetham, W.: Financial Application of Fuzzy Case-Based Reasoning to Residential Property Valuation. In Proceedings of the 6th IEEE International Conference on Fuzzy Systems, Vol. 1, pp. 37-44, (1997).
5. Bruner, J.: The Narrative Construction of Reality. Critical Inquiry, 18, pp. 1-21, (1991).
6. Finnie, G., Sun, Z.: Similarity and Metrics in Case-based Reasoning. Intelligent Systems, 17(3), pp. 273-285, (2002).
7. Gomide, F., Nakamiti, G.: Fuzzy Sets in Distributed Traffic Control, 5th IEEE International Conference on Fuzzy Systems - FUZZ-IEEE 96, pp. 1617-1623, New Orleans - LA - EUA, (1996).
8. Hansen, B.K., Riordan, D., Weather Prediction Using Case-Based Reasoning and Fuzzy Set Theory, Workshop on Soft Computing in Case-Based Reasoning, 4th International Conference on Case-Based Reasoning (ICCBR01), Vancouver, (2001).
9. Kleiner, A. and Roth, G., How to Make Experience Your Companys Best Teacher, Harvard Business Review, Vol. 75, No. 5, p 172, (1997).
10. Kolodner, J., Case-Based Reasoning, Morgan Kaufmann, San Mateo (CA), (1993).
11. Petraglia, F. and Sartori, F., Exploiting Artificial Intelligence Methodologies to Support Learning by Doing Within Organisations., In G. Hawke and P. Hager (eds.): Proceedings of RWL405-The 4th Int. Conference on Researching Work and Learning, Sydney, December 2005, (2005).
12. Wenger, E., (1998) Community of practice: Learning, meaning and identity, Cambridge University Press, Cambridge, MA, (1998).

Knowledge Modeling Framework for System Engineering Projects

Olfa Chourabi[1], Yann Pollet[2], and Mohamed Ben Ahmed[3]

Abstract System Engineering (SE) projects encompass knowledge-intensive tasks that involve extensive problem solving and decision making activities among interdisciplinary teams. Management of knowledge emerging in previous SE projects is vital for organizational process improvement. To fully exploit this intellectual capital, it must be made explicit and shared among system engineers. In this context, we propose a knowledge modelling framework for system engineering projects. Our main objective is to provide a semantic description for knowledge items created and/or used in system engineering processes in order to facilitate their reuse. The framework is based on a set of layered ontologies where entities such as domain concepts, actors, decision processes, artefacts, are interlinked to capture explicit as well as implicit engineering project knowledge.

1. Introduction

System Engineering (SE) is an interdisciplinary approach to enable the realization of successful systems. It is defined as an iterative problem solving process aiming

[1] Olfa chourabi
CNAM, 292, rue saint martin, paris
RIADI, campus universitaire de la Manouba.
Olfa.chourabi@voila.fr

[2] Yann pollet
CNAM, 292, rue saint martin, paris. pollet@cnam.fr

[3] Mohamed Ben Ahmed
RIADI, campus universitaire de la Manouba, mohamed.benahmed@riadi.rnu.tn

Please use the following format when citing this chapter:

Chourabi, O., Pollet, Y. and Ahmed, M.B., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 183192.

at transforming user's requirements into a solution satisfying the constraints of: functionality, cost, time and quality. [1]
System engineering projects involve the definition of multiple artifacts that present different formalization degrees, such as requirements specification, system architecture, and hardware/software components. Transitions between the project phases stem from decision making processes supported both by generally available domain and design knowledge.
We argue that Knowledge about engineering processes constitutes one of the most valuable assets for SE organizations. Most often, this knowledge is only known implicitly, relying heavily on the personal experience background of system engineers. To fully exploit this intellectual capital, it must be made explicit and shared among project teams. Consistent and comprehensive knowledge management methods need to be applied to capture and integrate the individual knowledge items emerging in the course of a system engineering project. Knowledge management (KM) is a scientific discipline that stems from management theory and concentrates on the systematic creation, leverage, sharing and reuse of knowledge resources in a company. [2] Knowledge management approaches are generally divided into personalization approaches that focus on human resources and communication, and codification approaches that emphasize the collection and organization of knowledge [3].
In this paper, we only consider the latter approach. Special focus is put on the comprehensive modeling of system engineering project knowledge. This knowledge partly resides in the product itself, while a lot of different types of knowledge are generated during the engineering processes. The background information such as why engineers came up with the final shape or geometry, what constraints were to be considered in engineering processes, and so on, can not be found either [4]. In other words, most of design rationale either disappear or exist partially in the form of engineering documents. In such setting, the most critical issue is related to the construction of a structured representation for engineering project knowledge modeling that record engineers' ideas and reasoning processes for a specific issue. This representation must be based on a formal language with expressive semantics, in order to perform computable operations on the recorded knowledge items and to improve their retrieval. Ontological engineering [5], which is the successor to knowledge engineering, has been expected to resolve the problem of semantic based knowledge modeling. In the engineering domain, the typical expectations for ontologies are: interoperability among engineering supporting systems, semantic constraints for modeling, implicit knowledge capture and knowledge systematization, [6]. In the context of our research, we use the term "ontology" as a formal structure providing a basis of knowledge systematization. [6] We propose a set of layered ontologies for representing the relevant engineering project entities and their Knowledge Modeling Framework for System Engineering Projects.

2. Background and motivation

2.1 System Engineering processes

System engineering (SE) is an interdisciplinary approach to enable the realization of successful systems. It is defined as an iterative problem solving process aiming at transforming user's requirements into a solution satisfying the constraints of: functionality, cost, time and quality [1]. This process is usually comprised of the following seven tasks: State the problem, Investigate alternatives, Model the system, Integrate, Launch the system, Assess performance, and Re-evaluate. These functions can be summarized with the acronym SIMILAR: State, Investigate, Model, Integrate, Launch, Assess and Re-evaluate. [7]. This Systems Engineering Process is shown in Figure 1.

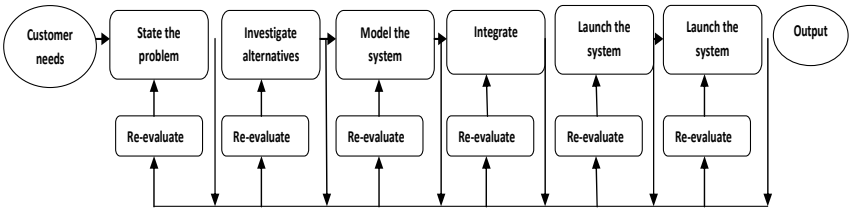


Fig 1: System Engineering Process [7]

It is important to note that the System Engineering Process is not sequential. Tasks are performed in a parallel and iterative manner. At each step a comprehensive set of possible engineering models arises witch are progressively combined and refined to define the target system.

Because of its inherent creative nature, it is a special case of business process. It is poorly structured and, as a rule, evolves in an unpredictable manner. In such highly dynamic settings with continuously changing requirements, the overwhelming majority of the engineering ways of working are not properly formalized, but are heavily based on the experience knowledge of the human performers.

As a consequence, engineering support environments have further to deal with the systematic collection of experience from previous project cycles and its dissemination and utilization from analogous problem solving contexts in the future. [8]. in section 3, we present a knowledge modeling framework that acts as a backend for what we expect to be a "Next generation of engineering support environment" i.e.: "knowledge centric" rather than "data centric" [12].

2.2 Knowledge Management issues in SE

The Above-delineated characteristics of SE processes show that a significant amount of knowledge is involved to solve a mix of ill- and well-defined problems. System engineers require topic knowledge (learned from text books and courses) and episodic knowledge (experience) [9].
One of the main problems in SE processes is the lack of capture and access to knowledge underpinning the design decisions and the processes leading to those decisions [10, 11].
System Engineers spend large portions of their time searching through vast amounts of corporate legacy data and catalogs searching for existing solutions which can be modified to solve new problems or to be assembled into a new device. This requires utilizing databases or online listings of text, images, and computer aided design (CAD) data. Browsing and navigating such collections are based on manually-constructed categorizations which are error prone, difficult to maintain, and often based on an insufficiently dense hierarchy. Search functionality is limited to inadequate keyword matching on overly simplistic attributes; it lacks the formal framework to support automated reasoning. [8]
In this paper, we focus on the knowledge modeling issue which is often considered as the first step in developing Knowledge-Based Systems (KBS). The aim of this process is to understand the types of data structures and relationships within which knowledge can be held, and reasoned with. We use ontologies to describe the knowledge model in a formal representation language with expressive semantics.
In order to determine the basic building blocks of the knowledge repository, we introduce the notion of “SE-Project Asset” as the smallest granularity in the system experience knowledge. “SE-Project Asset”, represent an integrated structure that capture product and process knowledge in engineering situations in conformance to set of layered ontologies.

3. Knowledge Modeling Framework for System Engineering Projects

In this section, our framework for knowledge modeling in system engineering projects is described. It structures the traces of engineering in the form of semantic descriptions based on a system engineering ontology. Section 3.1 introduces the so-called “SE general Ontology”, and Section 3.2 describes the modeling layers considered for semantic knowledge capture.

3.1 System Engineering General Ontology

Basically, our model aims at specifying explicitly the facets describing an "SE-Project Asset".
We choose to model these description facets with ontologies. In the knowledge engineering community, a definition by Gruber is widely accepted; that is, "explicit specification of conceptualization" [13], where conceptualization is "a set of objects which an observer thinks exist in the world of interest and relations between them" [14]. In engineering domain, ontology is considered "a system (systematic, operational and prescriptive definitions) of fundamental concepts and relationships which shows how a model author views the target world and which is shared in a community as building blocks for models. [6]
By instantiating these ontological concepts, concrete "SE-Project Asset" could be stored in a system engineering repository for future reuse. Furthermore, the ontology itself can serve as a communication base about the products and processes e.g. for exploring domain knowledge for system engineers.
We propose three description facets to capture the "SE-Project Asset". These three facets are arranged in a "SE general ontology" that introduces top-level concepts describing products and processes, as well as their interrelations and dependencies, independently from any particular engineering domain or application.

The main idea is to capture the engineering products, engineering processes, the design rationale, and the domain concepts in order to provide a comprehensive and computable description for projects knowledge. These descriptions facets are arranged around the "SE-Project Asset" as the central concept for SE project knowledge modeling.

-Domain facet: contains basic concepts and relations for describing the content of engineering assets on a high semantic level. It can be regarded as domain ontology for system engineering. In order to capture all engineering artifacts in a comprehensive manner, we propose to integrate in this facet a systematic description of: domain requirements, domain functions and behavior, domain architecture, and domain physical components. This decomposition constitutes typical system engineering modeling areas. Nevertheless, they could be extended or restricted in function of the engineering domain and the knowledge modeling scope. We work on aligning this domain facet with the reference device ontology described in [15]. Figure 3 presents a high level description of a typical domain facet.

-Product facet: contains concepts and relations representing artifact types as well as their information model. In SE domain, a system is described with several views such as: contextual, dynamic, static, functional or organic. By formally

relating modeling elements to domain concepts we could provide a systematic and semantic description of an engineering solution.

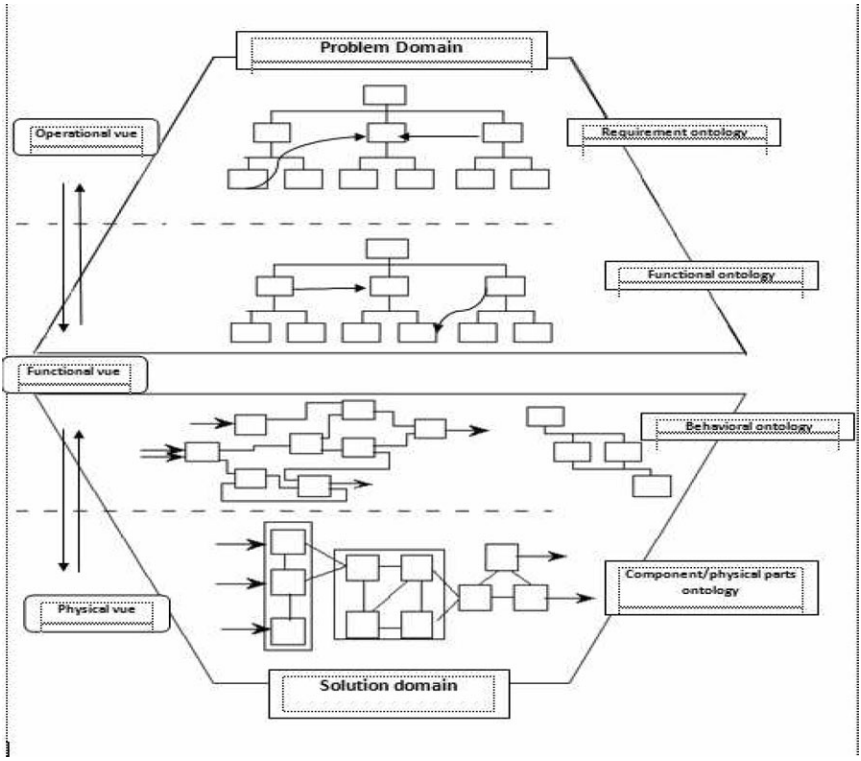


Fig 3: Ontologies for system engineering domain facets

-Process facet: contains concepts and relations that formally describe engineering activities, tasks, actors, and design rationales concepts (intentions, alternatives, argumentations and justification for engineering decisions). Both the process and the product facets act as a formal structure for the SE-Project Asset. The domain facet provides semantic domain values for characterizing this structure. Figure 4, illustrates the relationships and the complementarily of our three modeling facets for comprehensively representing SE-Project Assets.

3.2 Multi-layered Ontologies for SE Knowledge Modeling

For the sake of generality of our modeling framework, we have proposed the higher-level concepts for modeling SE-Project Asset. The concepts presented in the above section must be specialized and refined in order to provide operational

knowledge model for system engineering projects.

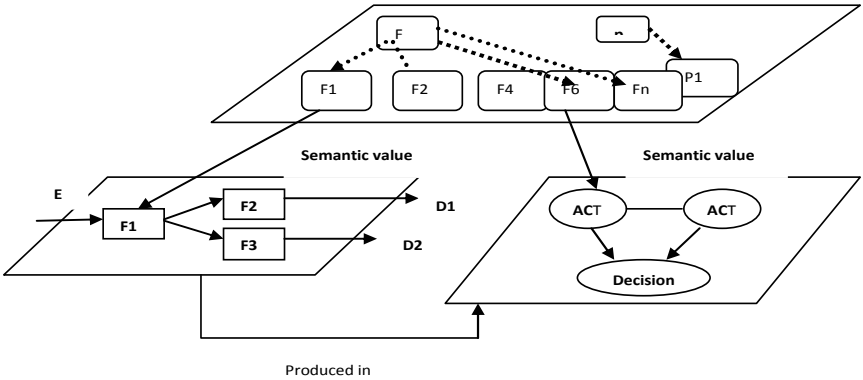


Fig 4: SE general ontology: domain, product and process facet

More precisely, the proposed SE General ontology must be refined in function of the engineering domain (aeronautics, information system, automobile etc.) and in function of the system engineering organizational context.

We propose an ontological framework organized into four semantic layers: layers subdivide the ontology into several levels of abstraction, thus separating general knowledge from knowledge about particular domains, organizations and projects.

Basically, knowledge in a certain layer is described in terms of the concepts in the lower layer.

Figure 5 shows a hierarchy of ontologies built on top of SE general ontology.

The first layer aims to describe super-concepts that are the same across all domains, it corresponds to the SE General ontology. The domain layer defines specializing concepts and semantic relations for a system engineering domain such as aeronautics. It integrates for examples domain theories and typical domain concepts that are shared in an engineering community. The application layer, presents specialized concepts used by specific system engineering organization, this is the most specialized level for knowledge characterization and acts as a systematized representation for annotating engineering knowledge projects. The fourth layer corresponds to semantic annotation on SE project assets defined using conceptual vocabulary from the application layer. In this way, all SE project assets are captured as formal knowledge models, by instantiating these ontological concepts.

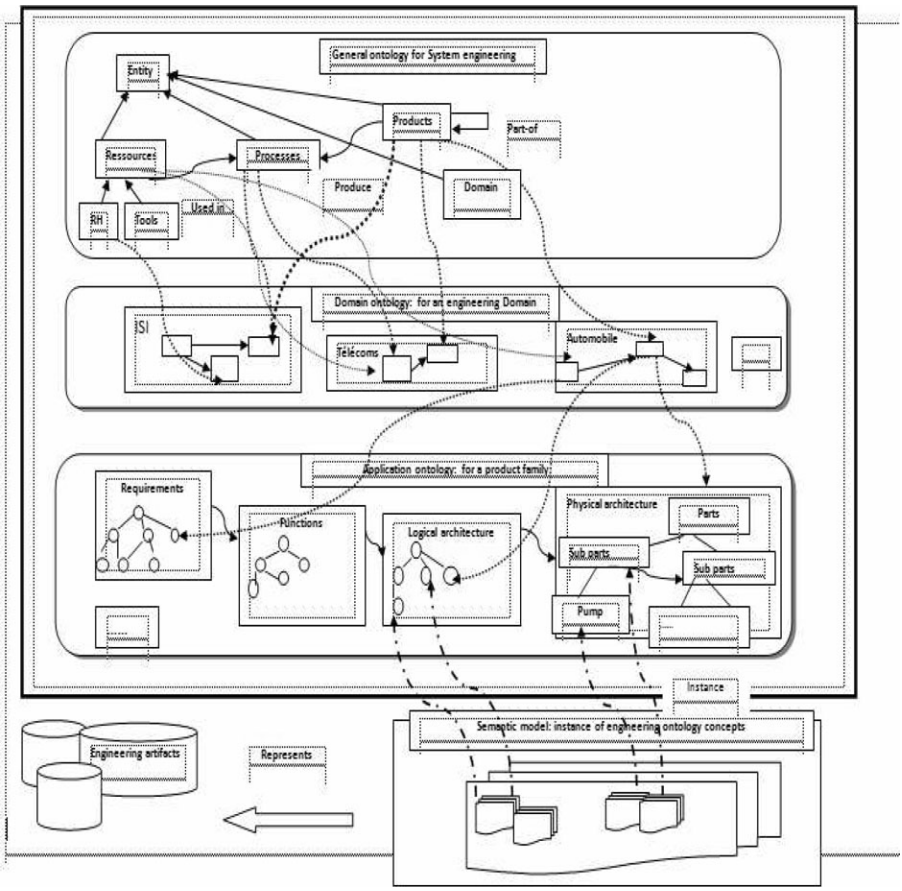


Fig 5: Ontology layers for SE projects

4. Related work

Most of the existing SE tools still lack essential aspects needed for supporting knowledge capitalization and reuse during projects processes. To our knowledge, there is no generic framework for knowledge management in SE domain.
In design engineering domain, [16] have integrated concepts of artificial intelligence into commercial PDM systems. The software is based on a dynamic and flexible workflow model, as opposed to the deterministic workflows seen in most commercial PDM applications. [17] Describes an integration of a PDM system with ontological methods and tools.

The Protégé ontology editor is combined with a commercial PDM system to provide knowledge management capabilities for the conceptual design stage. [18] Have designed ontology for the representation of product knowledge. A Core Ontology defines the basic structure to describe products from a functional view. An ontological architecture for knowledge management that resembles to our proposed framework has been proposed by Sebastian C. Brandt [19] and illustrated in the context of chemical engineering processes. Our knowledge modeling approach, in a way, tries to combine and to extend the ideas underlying the discussed related works into a coherent framework and to tailor them towards the specific system engineering domain and application.

5. CONCLUSION

System engineering processes implies the management of information and knowledge and could be considered as a knowledge production process. We have proposed a knowledge modeling framework based on ontologies for capturing in a semantic level the informal engineering artifacts in SE projects. A principal strand of future research is the application of this modeling framework in the context of an engineering organization to trigger further improvement. We plan also to use the same framework for capturing "best practices" knowledge. The problem of providing a knowledge management interface integrated to existing system engineering support tools is also under investigation.

References

1. J. Meinadier. Le métier d'intégration de systèmes. Hermes Science Publications, décembre 2002.
2. Awad, E. M., & Ghaziri, H. M. Knowledge management. Prentice Hall. 2003.
3. McMahon, C., Lowe, A., & Culley, S. J. (2004). Knowledge management in engineering design, personalisation and codification. Journal of Engineering. Design, 15(4), 307-325.
4. Chan-Hong park, M. Kang, J. Ahn, A knowledge representation framework in design repositories. Solid state phenomena Vol 120 (2007), pp. 235-240.
5. Riichiro Mizoguchi, Yoshinobu Kitamura. "Foundation of Knowledge Systematization: Role of Ontological Engineering". The Institute of Scientific and Industrial Research, Osaka University, 8-1, Mihogaoka, Ibaraki, Osaka, 567-0047, Japan.
6. YOSHINOBU KITAMURA. "Roles Of Ontologies Of Engineering Artifacts For Design Knowledge Modeling". In Proc. of the 5th International Seminar and Workshop Engineering Design in Integrated Product Development. (EDIProD 2006), 21-23 September 2006, Gronów, Poland, pp. 59-69, 2006C.
7. A. T. Bahill and B. Gissing, Re-evaluating systems engineering concepts using systems thinking, IEEE Transaction on Systems, Man and Cybernetics, Part C: Applications and Reviews, 28 (4), 516-527, 1998.

8. Michalis Miatidis, Matthias Jarke. “Toward Improvement-Oriented Reuse of Experience in Engineering Design Processes”. R. Khosla et al. (Eds.): KES 2005, LNAI 3684, pp. 228-234, 2005.
9. Robillard, P.N., The role of knolwedge in software development. Communications of the ACM, 1991. 42(1): p. 87-92.
10. Ali-Babar, M., I. Gorton, and B. Kitchenham, A Framework for Supporting Architecture Knowledge and Rationale Management, in Rationale Management in Software Engineering, A. H. Dutoit, et al., Editors. 2005,
11. Bosch, J. Software Architecture: The Next Step. European Workshop on Software Architecture. 2004.
12. Yann pollet. Intégration de systèmes à logiciel prépondérant Groupe « Modélisation des systems complexes » 22/10/2007, Conservatoire National des Arts et Métiers Chaire d’intégration des systemes.
13. Gruber, T. R., What is an Ontology?, http://wwwksl stanford.edu/kst/what-is-an-ontology.html
14. Genesereth, M. and Nilsson, N. Foundation of Artificial Intelligence, 1987.
15. de Kleer, J., Brown, J. S.: A Qualitative Physics Based on Confluences, Artificial Intelligence, Vol.24, 1984, pp.7-83.
16. Kim, Y., Kang, S., Lee, S., & Yoo, S. (2001). A distributed, open, intelligent product data management system. International Journal of Computer Integrated Manufacturing, 14, 224-235.
17. Gao, J. X., Aziz, F. L., Maropoulos, P. G., & Cheung, W. M. (2003). Application of product data management technologies for enterprise integration. International Journal of Computer Integrated Manufacturing, 16(7-8), 491-500.
18. Kopena, J., & Regli, W. C. (2003). Functional modeling of engineering designs for the semantic web. IEEE Data Engineering Bulletin, 26(4), 55-61.
19. Sebastian C. Brandt, Jan Morbach, Michalis Miatidis, Manfred Theißen, Matthias Jarke, Wolfgang Marquardt. An ontology-based approach to knowledge management in design processes. Computers and Chemical Engineering (2007).

FOUNDATIONS

Machines with good sense: How can computers become capable of sensible reasoning?

Junia C. Anacleto, Ap. Fabiano Pinatti de Carvalho, Eliane N. Pereira, Alexandre M. Ferreira, and Alessandro J. F. Carlos

Abstract Good sense can be defined as the quality which someone has to make sensible decisions about what to do in specific situations. It can also be defined as good judgment. However, in order to have good sense, people have to use common sense knowledge. This is not different to computers. Nowadays, computers are still not able to make sensible decisions and one of the reasons is the fact that they lack common sense. This paper focuses on OMCS-Br, a collaborative project that makes use of web technologies in order to get common sense knowledge from a general public and so use it in computer applications. Here it is presented how people can contribute to give computers the knowledge they need to be able to perform common sense reasoning and, therefore, to make good sense decisions. In this manner, it is hoped that software with more usability can be developed.

1 Introduction

Until nowadays computer are not capable of understanding about ordinary tasks that people perform in their daily life. They cannot reason about simple things using good sense as a person can do, and, therefore, they cannot help their users as they could if they had the capacity of making good judgment about the users' needs.

Since late 1950s, Artificial Intelligence (AI) researches have been looking for ways to make computers intelligent so that they could help their users in a better way. Part of those researchers believes that, in order to be intelligent, computers should first get the knowledge about human experiences, which involves knowledge about spatial, physical, social, temporal, and psychological aspects of typical everyday life. The set

Junia C. Anacleto, Ap. Fabiano Pinatti de Carvalho, Eliane N. Pereira, Alexandro M. Ferreira, Alessandro J. F. Carlos

Advanced Interaction Laboratory (LIA)
Computing Department - Federal University of São Carlos (DC/UFSCar)
Rod Washington Luis, Km 235 – São Carlos – SP – 13565-905
{junia, fabiano, eliane_pereira, alexandre_ferreira, alessandro_carlos}@dc.ufscar.br
+55 16 3351-8618

Please use the following format when citing this chapter:

Anacleto, J.C., de Carvalho, A.F.P., Pereira, E.N., Ferreira, A.M. and Carlos, A.J.F., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 195204.

of these kinds of knowledge, which is shared by most of people who have the same cultural background, is called common sense [1], [8], [11].

Actually common sense knowledge is very important to reach good sense because people use it to make their judgments. For example, in order to judge as wrong the attitude of a child badly responding to his parents, people have to consider some common sense facts such as "a child must respect older people", "parents are older than their children", "responding in a bad way is not respectful", and so on.

Since common sense is essential to reach good sense, how can this knowledge be provided to computers? One idea is to construct machines that could learn as a child, observing the real world. However, this approach was discarded after Minsky's and Papert's experience [15] of building an autonomous hand-eye robot, which should perform simple tasks like building copies of children's building-block structures. With this experience, they realized that it would be necessary innumerous short programs to give machines human abilities as cognition, perception and locomotion.

Another idea is to build a huge common sense knowledge base, to store it in computers and to develop procedures which can work on that knowledge. This seems an easier approach; nevertheless there are big challenges that must be won in order to get it [2], [7]. The first challenge of the second idea is to build the common sense knowledge base, since it is believed that to cover the human common sense it is necessary billions of pieces of knowledge such as knowledge about the world, myths, beliefs, etc. [1], [9], [10], and it is know that common sense is cultural dependent [1], [2]. Other challenges are presented in further sections.

Talking about building a large scale common sense knowledge base and developing applications capable of common sense reasoning, one attempt of this is the CYC Project, idealized by Douglas Lenat and that has been under development since 1984 [9]. In this project, Knowledge Engineers work on data that is gotten by interviewing people, and populate the project common sense knowledge base, storing the knowledge in a specific language, CycL. This approach has been proved to be very expensive, since nowadays the sum expended with the project exceeds tens of millions dollars [16]. In this way, CYC has been working on other alternatives [16].

Another attempt to build the desired common sense knowledge base and to use it in computer applications is the OMCS (Open Mind Common Sense) project [17], which takes into account the fact that every ordinary people has the common sense that computers lack and, so, everyone can help to construct the base. In this project the web technologies play a very important role in the knowledge base building. In order to get common sense facts it was developed web sites where anyone who knows how to write in a specific language – there are many versions of the project, each one in a language, such as English, Portuguese, Mexican and Korean – can subscribes himself and contribute by entering statements in natural language that originates a semantic network which is used by computer application.

This paper focuses on the OMCS-Br (Brazilian Open Mind Common Sense) project and its approaches to give common sense to computers. It is organized as follows: section 2 goes over some challenges that were faced since OMCS-Br has been under development at the Advanced Interaction Laboratory (LIA) from Federal

University of São Carlos, Brazil; section 3 brings some accomplishments of the project; and section 4 presents some conclusions and points to future works.

2 Challenges of getting and using common sense

Providing computers with common sense knowledge is an old dream of some AI researchers. In 1959, McCarthy was already concerned about the importance of giving this kind of knowledge to machines in order to make them intelligent [14]. Actually, there are those, as Marvin Minsky, who believes that the true intelligence with which computers should be supplied, lays on this kind of knowledge [15]. In spite of that, few projects have been developed to the purpose of reaching this dream. This is because there are difficult issues to deal with, as the ones experienced by OMCS-Br and explored in this paper.

First of all, to build a robust inference system based on common sense knowledge, it is necessary to construct a huge knowledge base [9], [10], [7]. However, what can be considered a huge knowledge base? Concerning those projects that collect common sense using natural language sentences, how many sentences should be necessary to cover the whole human knowledge? Furthermore, since it is known that common sense changes as time goes by, how long does it take to build the desired knowledge base? These are some questions that still have no answers, and maybe it was what has leaded some AI researchers not to invest on building common sense knowledge bases.

Nevertheless, OMCS-Br researchers believe that, whereas there is still no huge common sense knowledge base, it is possible to make machines more helpful and quite intelligent with a bit of common sense knowledge gotten from web contributors, as [10] showed to be possible. However, other questions rise up, as the ones related to the knowledge base quality. How can collaborators be guided in order to enter sentences related to the several kinds of knowledge which compose people's common sense? How should redundancy be treated? What about orthographic mistakes? A last question concerning the knowledge base construction: how can users be motivated to contribute on the website?

Now talking about the knowledge pre-processing which is necessary in order to use the acquired knowledge in computer application [9] [11], natural language has several syntactic structures. How should sentences be managed in order to get a better use of the knowledge they express? Which natural language parser should be used?

These are some questions which OMCS-Br has faced with since it has been launched. The approaches adopted by the project to some of them are presented in the next section.

3 The OMCS-Br Project accomplishments

Notwithstanding challenges, OMCS projects have been working to overcome all of them. Here it is presented how OMCS-Br has been approaching some issues previously mentioned. To begin with the knowledge base building, OMCS-Br

adopts template-based activities which guide users in such a way that they can contribute with different kinds of knowledge. The templates are semi-structured sentences in natural language with some lacunas that should be filled out with the contributors' knowledge so that the final statement corresponds to a common sense fact. They were planned to cover those kinds of knowledge previously mentioned and to get pieces of information that will be used further to give applications the capacity of common sense reasoning. The template-based approach makes easier to manage the knowledge acquired, since the static parts are intentionally proposed to collect sentences which can be mapped into first order predicates, which composes the project's semantic network. In this way, it is possible to generate extraction rules to identify the concepts present in a statement and to establish the appropriate relation-type between them. In OMCS projects, there are twenty relation-types, used to represent the different kinds of common sense knowledge, as it is presented in [11].

Those templates have a static and a dynamic part. The dynamic part is filled out by a feedback process that uses part of sentences stored in the knowledge base of the project to compose the new template to be presented. Figure 2 exemplifies how the feedback process works. At the first moment the template "You usually find a _____ in a **chair**" of the activity *Location* is presented to a contributor – the templates bold part is the one filled out by the feedback system. In the example, the contributor fills out the sentence with the word "*screw*". Then, the sentence *"You usually find a screw in a chair"* is stored in the OMCS knowledge base. At the second moment, the template "A **screw** is used for _____" of the activity *Uses* is shown to another contributor. Note that the word *screw* entered at the first moment is used to compose the template presented at the second moment.

Figure 1. Example of the OMCS-Br feedback process

The feedback process used in OMCS-Br website was planned in order to allow varied templates to be generated so that users are able to contribute on several subjects and do not get bored with always filling out the same sentence.

Still related to the feedback process, considering that the sentences stored in the knowledge base will be used to compose templates that will be shown to other contributors, it is important to provide a way through what it could be selected the

sentences that should be used by the feedback process. Thinking in this need, it was developed in OMCS-Br an on-line review system, which can be accessed just by the ones who have administrator privileges, where the sentences are selected to be or not to be used by the feedback process.

In order to perform the review, it was defined some rules to assure that common sense knowledge would not be discarded. The rules adopted in the review process are the following:

1. Sentences generated from a template that was filled out with a set of character without any meaning in Brazilian Portuguese are rejected. For example – if someone fills out a template with "*dafasdfasd*" the sentence is rejected;
2. Sentences with errant spelling, e.g., sentences that were filled out with words that are written orthographically wrong, are rejected;
3. Sentences generated by a template which was filled out differently from the default defined by the Knowledge Engineers to that activity, are accepted, but the entry is not used in feedback process. This happened for example, when the Knowledge Engineer defined that the default entry to a template is a noun phrase but the contributor filled it out with a verbal phrase. The entry is accepted, if all words are orthographically correct. The justification to this approach is that if the entry is accepted to the feedback process, it will be generated templates syntactically incorrect.
4. Sentences generated by a template that was filled out with bad words are accepted, but this entry is not used by the feedback process.

It is worth pointing out that during the review process the reviewer is not allowed to judge the semantic of a sentence. That is because it does not matter if a sentence seems strange in meaning or if it has already been scientifically proved as wrong. Common sense knowledge does not match scientific knowledge necessarily. Since a sentence is accepted as true by the most people who share the same cultural background, it is considered as a common sense sentence. Because of that reviewers are not allowed to judge if a sentence is common sense sentence or not.

Besides the templates about general themes such as those about *"things"* which people deal with in their daily life, *"locations"* where things are usually found and the common *"uses"* of things, there are also, in the Brazilian project website, templates about three specific domains: **health**, **colors** and **sexual education**. They are domains of interest to the researches that are under development in the research group which keeps the project [5] [4] [2]. This approach is only used in Brazil and it was adopted taking into account the necessity of making faster the collection of common sense knowledge related to those domains. The specific-domain templates were defined with the help of professionals of each domain. They were composed with some specifics words which instantiate the templates of general themes, in order to guide users to contribute with sentences related to a domain. Table 1 shows the accomplishments that OMCS-Br has gotten with that approach.

Table 1. Contributions on specific domains in OMCS-Br

Domain	Number of contributions	Period of collection
Health	6505	about 29 months
Colors	8230	about 26 months
Sexual Education	3357	about 21 months

The numbers of contributions in each domain can seem to be irrelevant, however, considering the only 2 facts about AIDS found in the knowledge base before creating the theme Sexual Education, it can be noticed the importance of domain-contextualized templates in order to make fast the collection of statements related to desired domains.

Another accomplishment of the OMCS-Br is related to the variety of contributor profiles. Nowadays there are 1499 contributors registered in the project site of which 19.33% are women and 80.67% are men. The most part of contributors (72.80%) is from Brazil South-east area, followed by the South area (15.25%). Those numbers point to the tendency that is proved by geographic sciences, which present the South-east and South area as being the most developed areas of Brazil. Considering that, it is perfectly understandable that, being well developed areas, their inhabitants have easier access to the Internet. Table 2 and Table 3 present other characteristics of OMCS-Br contributors.

Table 2. Percentage of contributors by age group

Age group	Percentage
Younger than 12 years	0.75 %
13 – 17	20.51 %
18 – 29	67.36 %
30 – 45	9.88 %
46 – 65	1.22 %
Older than 65 years	0.28 %

Table 3. Percentage of contributors by school degree

School degree	Percentage
Elementary school	2.21 %
High school	18.17 %
College	65.86 %
Post-Graduation	4.52 %
Master Degree	7.04 %
Doctorate Degree	2.21 %

Another conquest of OMCS-Br is the amount of contributions. Within two years and a half of project, it was gotten more than 174.000 sentences written in natural language. This was possible thanks the web technology and the marketing approach adopted by LIA. As the project was released in Brazil in 2005, it was realized that the knowledge base would rise up significantly just when there were an event that put the project in evidence. Figure 2 demonstrates this tendency.

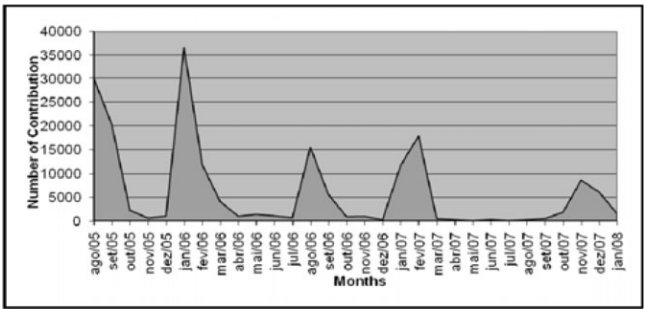


Figure 2. OMCS-Br knowledge base tendency of growing up

It can be noticed in Figure 2 that the periods where the knowledge base grew up significantly were from August to October 2005, from January to March 2006, from August to October 2006, from January to February 2007 and from November to December 2007. This is an interesting fact, because those jumps in the knowledge base just followed some marketing appeals performed by LIA. In the first one, LIA got

published some articles in some newspapers of national coverage telling people about the project and asking for people contribution. After had those articles printed, the OMCS-Br knowledge base reach the number of 50.000 sentences. Three months later, the knowledge base established and passed to grow up very slowly.

Thinking of having another jump in the knowledge base size, it was released in the later January 2006 a challenge associated to the Brazilian carnival. In that challenge, it was offered little gifts as prizes to the three first collaborators that contributed with the most number of sentences in the site activities. The winners received T-Shirts of the OMCS-Br Project and pens of MIT. The challenge was announced among the project contributors, which received an e-mail telling about it. The announcement was also posted in the Ueba website (www.ueba.com.br), a site of curiosities which target public is people interested in novelties. As it can be noticed, the knowledge base size had a jump as soon as the challenge was launched. The same approach was used in August 2006, January 2007 and December 2007.

Although the approach has gotten a good response from the contributors in the first three challenges, it can be noticed in Figure 2 that this approach is becoming inefficient. Thinking about keeping the knowledge base growing up, it is under development some games, following project contributors' suggestions, in order to make the collection process funnier and more pleasant.

Besides the knowledge base growth, another important issue in OMCS-Br is the pre-processing of the sentences stored in the knowledge base. As the knowledge is collected in natural language, it might be put in a computational notation in order to be used in computer application. The knowledge representation adopted in OMCS-Br is a semantic network.

After being generated in the *extraction* process, i.e. the process which extracts the semantic network nodes from the natural language statements stored in the knowledge base and relates them through first order predicates, the network nodes are submitted to a *normalization* process. Since the sentences collected in the site can vary in their morphology, it is needed to manipulate those sentences in order to increase the semantic network connectivity.

In order not to have inflected concepts, which means same words varying in number, tense, etc, separated in the semantic network, a set of heuristics is applied on the contributions so that they are grouped in a single node of the semantic network.

The normalization process in OMCS-Br is performed using Curupira [13], a syntactic parser for Brazilian Portuguese. However, as the parser does not strip the sentence inflectional morphology, it was developed a Python module to normalize the nodes. For this purpose, it is used the inflectional dictionary developed in the UNITEX-PB Project [18], which has all inflectional forms of Brazilian Portuguese morphological classes.

The module works in 3 steps. First of all, each sentence token is tagged using the Curupira parser. Afterward, articles and cardinal numbers are taken off – proper nouns are kept in original form. Special Portuguese language structures are treats. For instance, the *ênclise* structure, which is a case of pronominal position where the pronoun is concatenated after the verb, is stripped from the sentences and the verb is put in the infinitive form. For example, the verb "*observá-la*" ("observe it") is normalized to "*observar*" ("to observe"). Overall, each tagged token is normalized by searching its normal form in the inflectional dictionary used. In this way, sentences that were separated by morphological variations, like "*comeria maçãs*" ("would eat

apples") and "*comendo uma maçã*" ("eating an apple"), are reconciled during the normalization process generating the normalized expression "*comer maçã*" ("to eat apple").

In order to check the connectivity of the network generated using and not using the normalization process a test was performed. The results of this measurement are presented in Table 4.

Table 4. Effects of the normalization process on the OMCS-Br semantic network structure

	non-normalized	normalized	normalized/ non-normalized
nodes	36,219	31,423	- 13.24 %
relations	61,455	57,801	- 5.95 %
average nodal edge-density	4.4643	3.3929	+ 31.57 %

These results can be interpreted as follows: regarding the number of nodes and relations, they were decreased after the normalization process. This confirms the tendency that the normalization process makes reconciliations between morphological variations, and thus unifies them.

Another result that can be inferred examining the connectivity of semantic network is that the nodal edge-density has increased more than 30%, which demonstrates that the normalization process improves the connectivity of nodes.

Other strategy to improve the connectivity of the network is to extract new relations from the original relations. This is made applying a set of heuristic inferences over the original relations nodes. The relations generated by these heuristics are K-Line relations, a kind of relation based on Minsk's Theory about the contextual mechanism in memory [15].

One of these heuristics identifies whether a node is composed by more than a word, finds the node components variations based on grammar patterns and establishes "ThematicKLine" relations between the variations which do not have any word in common. For example, in the node "*pote de mel na mesa*" ("honey jar on the table") it is found the following variations: "*pote de mel*" ("honey jar"), "*pote*" ("jar"), "*mel*" ("honey") e "*pote na mesa*" ("jar on the table"). So, it is generated the following ThematicKLine:

```
(ThematicKLine 'pote de mel' 'mesa')
(ThematicKLine 'pote na mesa' 'mel')
(ThematicKLine 'pote' 'mesa')
(ThematicKLine 'pote' 'mel')
(ThematicKLine 'mesa' 'mel')
```

Another heuristic considers the nominal adjuncts in a node. In Portuguese, the nominal adjunct is a phrase accessory term that delimits or specifies a noun, and can be composed by a noun followed by an adjective. With this construction, it is created "SuperThematicKLine" relations, which establish generalization/specialization relation between the nodes. This relation links the entire structure to the stripped adjective structure. For example, from the expressions "*sala grande*" ("big room") it is created the following relations:

```
(SuperThematicKLine 'sala grande' 'sala')
(SuperThematicKLine 'sala grande' 'grande')
```

In this way, related terms are linked one another in the semantic network which became consequently more connected.

These are the approaches which are used by the OMCS-Br project. The next section presents some conclusions on providing common sense to computers so that they can make sensible reasoning and points to some projects which are under development using the architecture of this project.

4 Conclusions and Future works

This paper presented the approaches adopted by OMCS-Br to collect common sense knowledge from a general public and use it in computer applications. The project has been working on three fronts to make possible the development of applications which are capable of common sense reasoning. It is believed that, giving computer this ability is a step on getting machines which can act with good sense. In this way, it would be possible to construct applications which can support their users in a better way, offering a contextualized help, according to the common sense knowledge which the machines were provided with.

A research developed at LIA has pointed to the fact that OMCS-Br knowledge bases store cultural differences as it is presented in [1] and in [2]. As future work, it is intended to invest in the development of applications with intelligent interfaces. Those interfaces would take into account the cultural context, since it is known that cultural differences impacts directly in the user interface [12]. Considering common sense knowledge, applications could offer an interaction instantiated to that cultural background. Another research developed at the laboratory is related to using common sense knowledge to support teachers to plan learning activities [6]. It is being investigated, how common sense knowledge can be used by teachers in order to make them concerned about the previous knowledge of its students, about the misconceptions that should be approached during a learning activity, since common sense register myths, believes and procedures of the daily life, and so on. Also common sense reasoning has been integrated to *Cognitor*, an authoring tool developed at LIA, whose main purpose is to support the development of learning material to be delivery electronically [3]. There is another research related to how common sense reasoning can be used in the development of educational games [4], which allow teachers to use common sense knowledge in order to contextualize the learning process.

Actually, there are lots of challenges to be won in order to reach the dream of make machines capable of common sense reasoning and, consequently, good sense reasoning. The OMCS-Br group is concerned about the innumerous challenges which they might deal with and it has been looking for solutions that can lead to the success of the project as a whole.

Acknowledgments

We thank FAPESP and CAPES for partially support this project. We also thank NILC that made Curupira *dll* available to us.

References

1. Anacleto, J. C.; Lieberman, H.; Tsutsumi, M.; Neris, V. P. A.; Carvalho, A. F. P.; Espinosa, J.; Zem-Mascarenhas, S.; Godoi, M. S. Can common sense uncover cultural differences in computer applications?. In *Artificial Intelligence in Theory and Practice – WCC 2006*, vol. 217, Bramer, M. (Ed). Berlin: Springer-Verlag, 2006. p. 1-10.
2. Anacleto, J. C.; Lieberman, H.; Carvalho, A. F. P.; Neris, V. P. A.; Godoi, M. S.; Tsutsumi, M.; Espinosa, J.; Talarico Neto, A.; Zem-Mascarenhas, S. Using common sense to recognize cultural differences. In *Advances in Artificial Intelligence – IBERAMIA-SBIA 2006*, LNAI 4140, Sichman, J. S. et al. (Eds). Berlin: Springer-Verlag, 2006. p. 370-379.
3. Anacleto, J. C.; Carlos, A. J. F.; Carvalho, A. F. P. de; Godoi, M. S. Using Common Sense Knowledge to Support Learning Objects Edition and Discovery for Reuse. In *Proc of the 13th Brazilian Symposium on Multimedia and the Web (Webmedia 2007)*, 2007, Gramado – Brazil. New York: ACM Press, 2007. p. 290-297.
4. Anacleto, J. C.; Ferreira, A. M.; Pereira, E. N.; Carvalho, A. F. P. de; Fabro, J. A. Culture Sensitive Educational Games Considering Commonsense Knowledge. In *Proc. of 10th International Conference on Enterprise Information System* (ICEIS 2008), 2008, Spain. In press.
5. Carvalho, A. F. P. de; Anacleto, J. C.; Zem-Mascarenhas, S. Learning Activities on Health Care Supported by Common Sense Knowledge. In: *Proc of the 23rd Symposium on Applied Computing*, 2008, Fortaleza – Brazil. New York: ACM Press, 2008 p. 1385-1389.
6. Carvalho, A. F. P. de; Anacleto, J. C.; Zem-Mascarenhas, S. Planning Learning Activities Pedagogically Suitable by Using Common Sense Knowledge. In *Proc of the 16th International Conference on Computing (CIC 2007)*, 2007, Mexico City. p. 1-6.
7. Denison , D.C. Guess who's smarter. *Boston Globe Online*, 25 mar. 2003.
8. Lenat, D. B. CYC: a large-scale investment in knowledge infrastructure. *Communications of the ACM*, New York, v. 38, n. 11, Nov. 1995.
9. Lenat, D. B.; Guha, R. V.; Pittman, K.; Pratt, D.; Shepherd, M. Cyc: toward programs with common sense. *Communications of the ACM*, New York, v. 33, n. 8, aug. 1990.
10. Lieberman, H.; Liu, H. Singh, P.; Barry, B. Beating common sense into interactive applications. *AI Magazine*, v. 25, n. 4, p. 63-76, 2004.
11. Liu, H.; Singh P. ConceptNet: a practical common sense reasoning toolkit. *BT Technology Journal*, v. 22, n. 4, p. 211-226, 2004.
12. Marcus, A. Culture class vs. culture clash. *Interactions*, New York, v. 9, n. 3, p. 25-28, mar. 2002.
13. Martins, R. T.; Hasegawa, R.; Nunes, M.G.V. Curupira: a functional parser for Brazilian Portuguese. In *Proc of the 6th Computational Processing Of The Portuguese Language, International Workshop (PROPOR 2003)*, LCNS 2721, Mamede, N. J et al. (Eds). Berlin: Springer-Verlag, 2003.
14. McCarthy, J. Programs with Common Sense. In *Proc of the Teddington Conference On Mechanization Of Tought Processes*, 1959.
15. Minsky, M. *The Society of Mind*. New York: Simon and Schuster, 1986.
16. Panton, K.; Matuszek, C.; Lenat, D. B.; Schneider, D.; Witbrock, M. Siegel, N. Shepard, B. Common Sense Reasoning – From Cyc to Intelligent Assistant. In *Ambient Intelligence in Everyday Life*. New York: Spring-Verlag, LNAI 3864, 2006. pp. 1-31.
17. Singh, P. The OpenMind Common sense project. *KurzweilAI.net,* 2002.
18. UNITEX-PB– http://www.nilc.icmc.usp.br:8180/ unitex-pb/index.html

Making Use of Abstract Concepts–Systemic-Functional Linguistics and Ambient Intelligence

Jörg Cassens and Rebekah Wegener

Abstract One of the challenges for ambient intelligence is to embed technical artefacts into human work processes in such a way that they support the sense making processes of human actors instead of placing new burdens upon them. This successful integration requires an operational model of context. Such a model of context is particularly important for disambiguating abstract concepts that have no clear grounding in the material setting of the work process. This paper examines some of the strengths and current limitations in a systemic functional model of context and concludes by suggesting that the notions of instantiation and stratification can be usefully employed.

1 Introduction

The exhibition of intelligent seeming behaviour is necessary for an artefact to be considered intelligent. Intelligent seeming behaviour is generally considered to be behaviour that is contextually appropriate. An ability to accurately read context is important for any animal if it is to survive, but it is especially important to social animals and of these perhaps humans have made the most out of being able to read context, where such an ability is tightly linked to reasoning and cognition [1].

The necessity of exhibiting some kind of intelligent behaviour has lead to the developments jointly labelled as *ambient intelligence* [2]. But to successfully create intelligent artefacts, the socio-technical processes and their changes through the use of mediating artefacts have to be examined more closely. This paper focuses on

Jörg Cassens
Department of Computer and Information Science, Norwegian University of Science and Technology, 7491 Trondheim, Norway e-mail: cassens@idi.ntnu.no

Rebekah Wegener
Centre of Language in Social Life, Department of Linguistics, Macquarie University NSW 2109, Australia e-mail: rwegener@ling.mq.edu.au

Please use the following format when citing this chapter:

Cassens, J. and Wegener, R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 205–214.

how a social-semiotic theory of language, in which context is seen as integral to understanding communication, can be usefully employed in ambient intelligence. Ambient intelligence and its requirements from semiotics is further discussed in section 2 below.

Semiotics, or the study of sign systems, is here examined using a systemic functional model (see section 3). Systemic functional linguistics is a social semiotic theory of language which treats all behaviour as meaning bearing. This includes the behaviour of non-human participants and is oriented to the shared rather than the unique aspects of sign systems. The relationship between semiotics and ambient intelligence is outlined in section 4 below. In this paper we discuss how a systemic functional approach to semiotics is valuable in defining abstract concepts, see section 5. Abstract concepts, or concepts which have no direct referent in the material setting, are an important part of the mental tool set for humans. They allow us to transcend the here and now by providing us with a shorthand for complex events or complex sets of ideas. Despite this benefit, they do represent a challenge for modelling within ambient intelligence. Because they lack a clear material referent, abstract concepts are difficult to disambiguate and respond appropriately to. We propose that a systemic functional model of context will be beneficial in understanding abstract concepts.

We conclude this paper by pointing to future work in this area. For example, while we have focused on devices designed to interact closely with a single user, humans typically interact in groups, so it will be necessary to consider the impact of this for environments where not all users share the same meaning system.

2 Ambient Intelligence

In understanding human cognition and reasoning, disciplines such as neuroscience, psychology, sociology, linguistics, and philosophy have had to take a stance on context as a concept. Setting aside the more mechanistic views taken on reasoning, which typically need not consider context at all, positions on context tend to fall into two broad domains: those who see context as vast and unable to be coded and those who view some form of generality and coding as being possible.

For social and practical reasons, historically, AI has drawn heavily from formal logic. For example, one of the benefits of such models was that they were comparably easy to implement. Formal logic is concerned with the explicit representation of knowledge and places great emphasis on the need to codify all facts that could be of importance. This focus on knowledge as an objective truth can be traced back to e.g. the logic of Aristotle who believed that at least a particular subset of knowledge had an objective existence (Episteme) [3]. This view contrasts with that of, for example, Polanyi, who argues that no such objective truth exists and all knowledge is at some point personal and hidden (tacit) [4].

The total denial of the existence of an objective truth is problematic, since consequently there can exist no criterion to value any representation of knowledge. We

can contrast this with the view of Kant, who regards the accordance of the cognition with its object as being presupposed in the definition of truth [5, p. 52]. Going further, he makes clear that a purely formal and universal criterion of truth cannot exist. He foregrounds the dialectic relation between the formal logic and the objects to which this logic may be applied and which are given through intuition. Such a dialectic approach overcomes the conceptual difficulties outlined above, but the consequences for computational models are not easily accounted for.

Context does not fit very well with the strict logical view on how to model the world. However, an extrememly personal and unique account of context serves little purpose in attempting generality. Context is, after all, a shared and very elusive type of knowledge. Despite the fact that humans can quite easily read context, context is hard to quantify in any formal way, and it is difficult to establish the type of knowledge that is useful in any given situation. Ekbia and Maguitman argue that this has led to context being largely ignored by the AI community [6]. Neither the relativist nor the formal logic approach to context has been very useful at producing accounts of context which resonate with the AI community, and, except for some earlier work on context and AI, Ekbia and Maguitman's observation still holds. Systemic-functional linguistics as described in the following section employs a dialectic view on context, and therefore avoids the pitfalls of the formal logic as well as the relativistic approaches.

3 Semiotics

Understanding meaning making and meaning making systems is the domain of Semiotics. Semiotics is commonly understood to be the study of sign systems and we here make use of systemic functional linguistics which is a social semiotic[7]. Semiotics itself has a long history and its use in computer science is not new, even if not extensive. However, it is not our intention in this paper to review the body of work surrounding semiotics though we are mindful of the impact of this work on the field today, in particular the work of Saussure [8], Peirce [9] and Voloshinov [10]. For a comprehensive account of semiotics as it is applied to computing we recommend works such as Gudwin and Queiroz [11] (in particular Bøgh Andersen and Brynskov [12] and Clarke et al. [13]) as well as de Souza [14]. The intelligent artefacts that we consider in this paper are an integral part of social interaction. They change the sense making process on the side of the human users as well as their own functioning as signs (contextualised by the users). Ideally, the artefact should be able to adapt to its use and user, and the means for this adaptation will have to be laid out by the designers.

In this research, we have used the social semiotics outlined by Halliday (see for example [15] and [16]). Halliday combines the strengths of the approaches of Saussure, Pierce, and Voloshinov. He brings together the tradition of relational thinking from Saussure, the understanding that different modalities have consequences for

the structure of meanings from Pierce, and from Voloshinov, the insistence that the sign is social.

Halliday's Systemic Functional Theory of language (SFL) is a social semiotic theory that sets out from the assumption that humans are social beings that are inclined to interact [15]. In this paper we examine the value of the SFL notion of context, which views context as all the features of a social process relevant to meaning making. These features are organised into 3 core parameters of context: Field, Tenor and Mode, where **field** is *"the nature of the social activity..."*, **tenor** is *"the nature of social relations..."*, and **mode** is *"the nature of contact..."* [17]. Context, in SFL is one of four linguistic levels (see below), which are related realizationally rather than causally, meaning that patterns on one level both construe and construct patterns on another level. Halliday manages the complexity of language by modelling it as a multidimensional system. The most crucial dimensions of this multidimensional system for our purposes are: stratification and instantiation. We examine how these key notions of SFL make this model of context valuable for AI. Focusing in particular on the notion of instantiation.

Stratification: Halliday uses a stratified model of language that incorporates the levels of the expression plane (including sound systems – phonetics and phonology, gesture, pixels etc.), lexicogrammar (lexis/grammar – or wording and structure), semantics (the meaning system) and context (culture and situation – elements of the social structure as they pertain to meaning). Description on each stratum is functionally organised into systems.

Instantiation: Halliday uses a tripartite representation of language, which has language as system, language as behaviour and language as knowledge. Language as system encapsulates the abstract structure of language. This accounts for the regularised (though changeable) patternings that we see in language. It is this regularity that makes prediction and a certain degree of formalism (at least of a functional nature) possible. Language as behaviour looks at the activity of language, while language as knowledge looks at the way in which we know language. But we do not do these things independently. We do not know language as a set of abstract rules. Rather we know language in the sense of knowing how to use it, in the sense of knowing how to communicate with others [15]. In practice these things occur together. When we try to build a device, it is language behaviour and knowledge that we face, yet it is the seemingly inaccessible system that we need to encode in order to produce intelligent seeming behaviours and knowledge in the device.

The concept that encapsulates this problem is what Halliday calls the cline of instantiation. This is a way of looking at the relationship between System (which at the level of context means the culture) and Instance (which at the level of context means the situation that we are in). This is represented in figure 1. Here we see in the foreground the system view of language, and its grounding in the instance.

The formalization of a level of context as part of a polysystemic representation of language has long been emphasized in the work of systemic functional linguists, especially Halliday and Hasan [18]. It is the dialectic approach of systemic functional linguistics which avoids the problem of vastness and that of uniqueness.

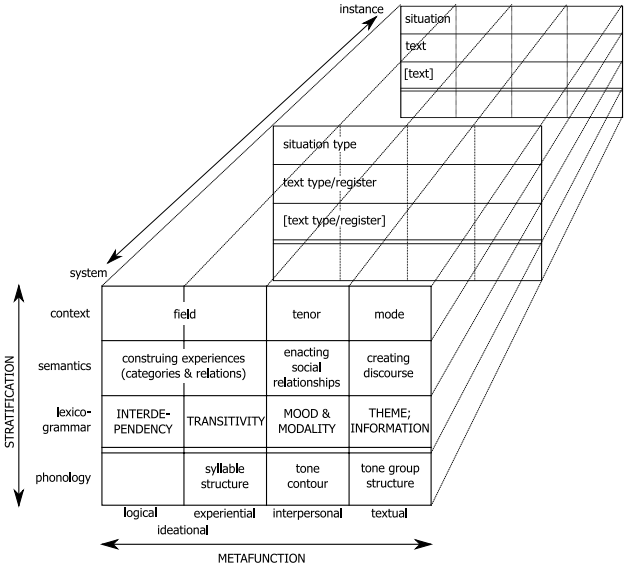


Fig. 1 The dimensions of language – Halliday and Matthiessen

Instances that share a similar function, e.g. instances of ward rounds in hospitals, typically share a similar structure. Halliday refers to these situation types as registers and they represent a functional variety of language [16]. The value of register is that we do not have to describe everything. Register can be thought of as an aperture on the culture. So, we are not faced with the full complexity of the culture. This does not mean that we do not keep the culture in mind. Any picture of a part of the system necessarily has the full system behind it. With register we set out from the instance, but keep in mind that each instance is a take on the system. Our notion of what constitutes an instance is shaped by our understanding of the culture/system. So, although Halliday represents the relationship between system and instance as a cline of instantiation, it is probably best understood as a dialectic since the two are never actually possible without each other. Register does not so much sit between system and instance, as it is a take on system and instance at the one time. It is the culture brought to bear on the instance of the social process.

For ambient intelligence, this means that we are not faced with the unhelpful uniqueness of each instance, because we are viewing it through the system and therefore foregrounding the shared aspects. Neither are we confronted with the seemingly impossible task of transcribing the infinity of culture, because we are viewing the culture through the aperture of the instance.

4 Semiotics in Ambient Intelligence

In this section, we will give our basic understanding of how semiotics can be used to understand the peculiarities of user interaction with ambient intelligent systems. The basic concept of the chosen interpretion of semiotics is the sign, a triadic relation of a signifier, a signified, and object. We look at the process of sense-making, where a representation *(signifier)* and its mental image *(signified)* refer to an entity *(object)* (the meaning of a sign is not contained within a symbol, it needs its interpretation).

On the background of semiotics, meaningful human communication is a sign process. It is a process of exchanging and interpreting symbols referring to objects. The user of a computer systems sees his interaction with this system on this background. When typing a letter, he does not send mere symbols, but signs to the computer, and the feedback from the machine, the pixels on the screen, are interpreted as signs: to the user, the computer is a "semiotic machine". The question that arises is whether a computer is actually itself taking part in the sense making process.

On one hand, following for example Kant, human understanding has as a necessary constituent the ability to conceptualise perceived phenomena through an active, discursive process of making sense of the intuitive perception [5, p. 58]. Following this understanding, computer systems are only processing signals, lacking the necessary interpreting capabilities humans have. They only manipulate symbols without conceptualising them.

On the other hand, we can take a pragmaticist approach, following for example Peirce and Dewey, and focus not on whether the machine is itself a sense maker, but on how its use changes the ongoing socio-technical process, and whether it can mediate the sense making process. From this point of view, the computer can be a sense making agent if its actions are appropriate in terms of the user's expectations.

Both approaches lead to a change in the issues we deal with when constructing an ambient intelligent system. The problem is transformed from one where the issue is to build a machine which itself realises a sense making process to one in which the issue is to build a computer thats actions are appropriate for the situation it is in and which exhibits sufficient sign processing behaviour.

We argue that, in order to make a pervasive, ambient intelligent system that behaves intelligently in a situation, it must be able to execute actions that make a difference to the overall sense making process in a given context. This differs from the interaction with traditional systems in which case the sense-making falls wholly on the side of the human user: You do not expect a text processor to understand your letter, but you expect an ambient intelligent system to display behaviour suggesting that it understands relevant parts of the situation you are in. When interacting with ambient intelligent systems, the user should be facilitated to subscribe to the sense making abilities of the artefacts. We consider the ability of the system to deal with concepts which have no direct material reference to be important to achieve this goal.

5 Abstract Concepts

Abstraction, or the ability to create a more general category from a set of specifics by whatever principle, is arguably one of the most useful mental tools that humans possess [19]. Indeed [20] suggests that the abstract categories that form part of our everyday life and language, are typically below conscious attention and only become apparent through linguistic analysis.

Such abstraction, though important to human intelligence, presents a challenge for modelling in ambient intelligence. Consider the meanings of the word 'Emergency'. Emergency has numerous meanings depending on the context in which it occurs. For the purposes of our discussion we will here limit ourselves to the hospital environment. In the hospital environment, 'emergency' has specific meanings that are distinct from the meanings in other contexts. Not only are there hospital specific meanings (culture specific), but the meaning varies according to the situation as well (situation specific). Within the hospital domain the term emergency may be understood to have two distinct meanings. Firstly, the term may mean the emergency department of the hospital. This is a concrete concept with a direct material referent of a place: the emergency department of the hospital. Drawing on the notion of stratification, we can see that this concept is typically realized in the lexicogrammar [1] by use of the specific deictic (e.g. 'the emergency department'), and by the possibility of using it as a circumstance location spatial (e.g. 'in the emergency department').

Secondly, the term may mean an emergency. This meaning of the term is an abstract concept with no direct referent in the material setting, referring instead to a state. This term is realized in the lexicogrammar by use of a non-specific deictic (e.g. 'an emergency') and may, if used in the past tense, use the specific deictic accompanied by a circumstance of location either spatial or temporal (e.g. 'the emergency in F ward' or 'the emergency this morning'). Note that here it is not the emergency that is the circumstance, but either time or location.

Our focus in this paper is on the second of these meanings. This meaning, an emergency, may be understood to refer to a complex set of actions and relations that constitute an interruption to the normal flow of a social process. This interruption may be:

- **Culture based:** deriving from the function of the broader hospital culture, or,
- **Context based:** deriving from variation within the structure of the social process itself.

It is this relation between culture based and context based meanings that is explored below.

To function intelligently in context, artefacts must be able to recognise 'emergency' and respond appropriately. They may need, for example, to "be quiet" while the doctor deals with an 'emergency' or they may need to "provide new information" needed by the doctor in an 'emergency'. To account for these complexities,

[1] This makes use of the relationship between patterns on different levels of language. For details, see section 3

a rich, but targeted, description of the culture is needed. To do this we will use the notions of register and generic structure potential [21] and a contextual model of language.

In order to establish what emergency means in this context we need to see its place in the system. That means we need to understand how it fits within the hospital culture. Understanding the richness of the culture is part of adequately embedding a device into that culture. Not doing so runs the risk of producing an artefact unsuited to its purpose and thus unintelligent. Part of what makes something (appear) intelligent is the ability to read and respond to the context. Context here is not just the immediate setting of the artefacts, (the context of situation), but the culture of which that setting is a part. Ward rounds then must be seen from the perspective of how they fit into the hospital culture. Within the function of the hospital, which is the restoration of health, the function of ward rounds is to monitor health. Because it has a 'monitoring' function within the hospital culture, it will be possible for the ward round to be interrupted by 'emergencies' from the wider hospital, since the function of the hospital overrides that of the ward round in terms of urgency.

By understanding the function of the ward round, and its contextual configuration, it is possible to state a generic structure potential for a ward round. A generic structure potential is a statement of the likely structure of a context. A generic structure however does not mean that there will not be variation.The notion of a ward round for example, is itself a functional abstraction[2] of all the behaviours, relations, and communications that go into completing a ward round. We are able to recognise from experience that certain behaviours by different participants, combined with certain roles and relations (e.g. ward doctor, ward nurse, patient, specialist) combined with the exchange of certain types of information (receiving information, requesting information, giving information) together constitute a ward round. None of these behaviours, relations or communications on their own constitutes a ward round, the ward round is identified by all of these things together.

Understanding the function both of the hospital within society and the ward round within that environment, facilitates the construction of a picture of the generic structure of a ward round and its place within the broader hospital culture. This enables a better understanding of the likely meaning of abstract concepts such as 'emergency'. Based on these conceptions of the ward round context, it is possible to posit the existence of two broad categories of emergency: those constituting an interruption to the ward round (when the hospital culture impinges on the ward round) and those constituting a change to the ward round (when there is internal variation in the ward round context). Because the first involves changes to the **field** (a new topic, ward, and focus), **tenor** (very different participants and role relations), it is likely to require a "new information response". This is because the field, tenor and mode settings have changed so much that it is now a new context and will thus require different information to suit this new context. The second will not involve changes to the **mode** or **tenor**, and only minor changes to the **field**. Thus it is likely to require a "be quiet and await query" response. This is because this is not a new context, it

[2] Here used to refer to the means by which abstraction is made, i.e. by considering the function of the behaviour.

is simply variation within the structure of the ward round. By utilising the notion of register to limit what we have to consider in the culture, and the concept of generic structure potential to model a typical view of the situation based on our study of the instances, we are able to better understand the context of the ward round and how to model abstract concepts for this context.

6 Conclusion and further work

In this paper we have considerered one of several ways that semiotics can be made fruitful in ambient intelligence. This research has suggested many areas of future investigation. In this project we have focused on the individual, but the sign making process is a negotiated process. It is not simply one meaner that has to be considered. In any exchange there are always at least two meaners, and more typically more than two. Multiparticipant communication represents a challenge to modelling. We have to keep in mind that others may share our conceptualisations and meanings only to a certain extent. When ambient intelligent systems link different people this is an important thing to remember. The closer a person is in our social network the more likely they are to share our meanings, while the further out in our social network the less likely they are to share meanings. In the hospital environment, ambient intelligent devices can belong to different groups of users. Should we model them in a way that the assistant of a nurse is more likely to share concepts with the assistant of another nurse than that of a physician?

Ambient intelligent systems will have deal with these kinds of challenges. Another point to consider is where in the network the system itself sits. What is the relation of the system to its user? To other pervasive devices? To their users? We are effectively dealing with a case of dialectal variation. Certain users may find some signs transparent and others not, while other users may find the exact opposite. If ambient intelligent systems are used to link people how do they best utilise signs to do this? This issue becomes very important when health care professionals from different cultural and language backgrounds have to interact.

Another issue we would like to explore further is the extent to which it is possible to relate a semiotic approach to ambient intelligent systems design to other socio-technical theories already in use in the field of ambient intelligence. A promising candidate is for example activity theory. Bødker and Andersen have outlined some properties of a socio-technical approach taking advantage of ideas from both theoretical frameworks [22], and we would like to extend this to cover specific aspects of SFL and Cultural-Historical Activity Theory (CHAT). This will potentially extend the number of projects from which we can borrow findings, meaning a richer description of the hospital environment.

Another point we have not fully explored yet is the relation of concepts from SFL with specific methods from the field of artificial intelligence. For example, the notion of genres in SFL seems to be a likely candidate for knowledge poor lazy learning mechanisms, while the descriptive power of register might be exploitable

in knowledge intensive or ontology based approaches. A promising candidate to combine these aspects is knowledge-intensive case-based reasoning.

References

1. Leake, D.B.: Goal-based explanation evaluation. In: Goal-Driven Learning. MIT Press, Cambridge (1995) 251–285
2. Ducatel, K., Bogdanowicz, M., Scapolo, F., Leijten, J., Burgelman, J.C.: ISTAG scenarios for ambient intelligence in 2010. Technical report, IST Advisory Group (2001)
3. Aristotle: Analytica Posteriora. Akademie (1998)
4. Polanyi, M.: Personal Knowledge: Towards a Post-critical Philosophy. N.Y.: Harper & Row (1964)
5. Kant, I.: Kritik der reinen Vernunft (2. Auflage). Akademie (1787)
6. Ekbia, H.R., Maguitman, A.G.: Context and relevance: A pragmatic approach. Lecture Notes in Computer Science **2116** (2001) 156–169
7. Hodge, R., Kress, G.: Social Semiotics. Cornell, UK (1988)
8. Saussure, F.d.: Course in General Linguistics. McGraw-Hill (1966)
9. Peirce, C.S.: New elements (kaina stoicheia). In Eisele, C., ed.: The New Elements of Mathematics by Charles S. Peirce. Volume 4, Mathematical Philosophy. (1904) 235–263
10. Voloshinov, V.N.: Marxism and the Philosophy of language. Seminar Press, New York (1973)
11. Gudwin, R., Queiroz, J.: Semiotics and Intelligent Systems Development. IGI Publishing, Hershey, PA, USA (2006)
12. Andersen, P.B., Brynskov, M.: The semiotics of smart appliances and pervasive computing. [11] 211–255
13. Clarke, R., Ghose, A., Krishna, A.: Systemic semiotics as a basis for an agent oriented conceptual modeling methodology. [11] 256–285
14. de Souza, C.S.: The Semiotic Engineering of Human-Computer Interaction (Acting with Technology). The MIT Press (2005) QA76.9.H85 .D465 2005.
15. Halliday, M.A.: Language as a Social Semiotic: the social interpretation of language and meaning. University Park Press (1978)
16. Halliday, M.A., Matthiessen, C.M.: An Introduction to Functional Grammar, Third edition. Arnold, London, UK (2004)
17. Hasan, R.: Speaking with reference to context. In Ghadessy, M., ed.: Text and Context in Functional Linguistics. John Benjamins, Amsterdam (1999)
18. Halliday, M.A., Hasan, R.: Language, Context, and Text: aspects of language in a scoial-semiotic perspective. Deakin University Pres, Geelong, Australia (1985)
19. Butt, D.: Multimodal representations of remote past and projected futures: from human origins to genetic counselling. In Amano, M.C., ed.: Multimodality: towards the most efficient communications by humans. 21st century COE Program International Conference Series NO. 6, Graduate school of letters, Nagoya University (2006)
20. Whorf, B.L.: Language, Thought and Reality (ed. J. Carroll). MIT Press, Cambridge, MA (1956)
21. Hasan, R.: Situation and the definition of genre. In Grimshaw, A., ed.: What's going on here? Complementary Analysis of Professional Talk: volume 2 of the multiple analysis project. Ablex, Norwood N.J. (1994)
22. Bødker, S., Andersen, P.B.: Complex mediation. Journal of Human Computer Interaction **20** (2005) 353–402

Making Others Believe What They Want

Guido Boella, Célia da Costa Pereira, Andrea G. B. Tettamanzi, and Leendert van der Torre

Abstract We study the interplay between argumentation and belief revision within the MAS framework. When an agent uses an argument to persuade another one, he must consider not only the proposition supported by the argument, but also the overall impact of the argument on the beliefs of the addressee. Different arguments lead to different belief revisions by the addressee. We propose an approach whereby the best argument is defined as the one which is both rational and the most appealing to the addressee.

1 A Motivating example

Galbraith [5] put forward examples of public communication where speakers have to address a politically oriented audience. He noticed how it is difficult to propose them views which contrast with their goals, values, and what they already know.

Speaker $\mathscr{S}$, a financial advisor, has to persuade addressee $\mathscr{R}$, an investor, who desires to invest a certain amount of money (*im*). $\mathscr{S}$ has two alternative arguments in support of a proposition *wd* ("The dollar is weak") he wants $\mathscr{R}$ to believe, one based on $bt \rightarrow wd$ and one on $hb \rightarrow wd$:

1. "The dollar is weak (*wd*) since the balance of trade is negative (*bt*), due to high import (*hi*)" ($a = \langle \{bt \rightarrow wd, hi \rightarrow bt, hi\} \rangle$)

Guido Boella
Universita di Torino, Dipartimento di Informatica 10149, Torino, Cso Svizzera 185, Italy, e-mail: guido@di.unito.it

Célia da Costa Pereira and Andrea G. B. Tettamanzi
Università degli Studi di Milano, Dip. Tecnologie dell'Informazione, Via Bramante 65, I-26013 Crema (CR), Italy, e-mail: {pereira,tettamanzi}@dti.unimi.it

Leendert van der Torre
Université du Luxembourg, Computer Science and Communication L-1359 Luxembourg, rue Richard Coudenhove – Kalergi 6, Luxembourg, e-mail: leon.vandertorre@uni.lu

Please use the following format when citing this chapter:

Boella, G., da Costa Pereira, C., Tettamanzi, A.G.B. and van der Torre, L., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 215224.

2. "The dollar is weak (*wd*) due to the housing bubble (*hb*) created by excess subprime mortgages (*sm*)" ($hb \rightarrow wd$, $sm \rightarrow hb$, *sm*). And to the reply of $\mathscr{R}$: "There is no excess of subprime mortgages (*sm*) since the banks are responsible (*rb*)" ($rb \rightarrow \neg sm$, *rb*), $\mathscr{S}$ counters that "The banks are not responsible (*rb*) as the Enron case shows (*ec*)" ($ec \rightarrow \neg rb$, *ec*).

Assume that both agents consider a supported proposition stronger than an unsupported one (e.g., $ec \rightarrow \neg rb$ prevails on *rb* alone). Although, from a logical point of view, both arguments make the case for *wd*, they are very different if we consider other dimensions concerning the addressee $\mathscr{R}$. For example, even if $\mathscr{R}$ could accept *wd*, other parts of the arguments have different impacts.

Accepting the arguments implies not only believing *wd*, but also the whole argument from which *wd* follows (unless we have an irrational agent which accepts the conclusion of an argument but not the reasons supporting the conclusion). This means that $\mathscr{R}$ undergoes a phase of belief revision to accept the support of the argument, resulting in a new view of the world. Before dropping his previous view of the world and adopting the new one, he has to compare them.

- The state of the world resulting from the revision is less promising from the point of view of the possibility for $\mathscr{R}$ of reaching his goals. E.g., if the banks are not responsible, it is difficult to achieve his goal of investing money *im*.
- The state of the world resulting from the revision contrasts with his values. E.g., he has a subprime mortgage and he does not like a world where subprime mortgages are risky due to their excess.
- He never heard about $hb \rightarrow wd$, even if he trusts $\mathscr{S}$; this is new information for him.

Thus $\mathscr{R}$ is probably leaning to accept the first argument which does not interact with his previous goals and beliefs, rather than to accept the second one, which, above all, depicts a scenario which is less promising for his hopes of making money by investing. Thus, a smart advisor, which is able to figure out the profile of the investor, will resort to the first argument rather than to the second one.

Even if such evaluation of $\mathscr{R}$'s in deciding what to believe can lead to partially irrational decisions, this is what happens in humans. Both economists like Galbraith and cognitive scientists like Castelfranchi [8] support this view. Thus, $\mathscr{S}$ should take advantage of this mechanism of reasoning.

In particular, an agent could pretend to have accepted the argument at the public level, since he cannot reply anymore to the persuader and he does not want to appear irrational. However, privately, and in particular when the time comes to make a decision, he will stick to his previous beliefs.

For this reason, if we want to build agents which are able to interact with humans, or believable agents, or if we want to use agent models as formal models for phenomena which are studied informally in other fields like economics, sociology, and cognitive science, and, moreover, to avoid that our agents are cheated by other agents which exploit mechanisms like the one proposed here, these phenomena must be studied.

2 Argumentation Theory

We adopt a simple framework for argumentation along the lines of Dung's original proposal [4] by instantiating the notion of argument as an explanation-based argument. Given a set of formulas L, an argument over L is a pair $A = \langle H, h \rangle$ such that $H \subseteq L$, H is consistent, $H \vdash h$, and H is minimal (for set inclusion) among the sets satisfying the former three conditions. On the set of arguments $\mathtt{Arg}$, a priority relation $\succeq$ is defined, $A_1 \succeq A_2$ meaning that A_1 has priority over A_2.

Let $A_1 = \langle H_1, h_1 \rangle$ and $A_2 = \langle H_2, h_2 \rangle$ be two arguments. A_1 undercuts A_2, in symbols $A_1 \rightsquigarrow A_2$, if $\exists h'_2 \in H_2$ such that $h_1 \equiv \neg h'_2$. A_1 rebuts A_2, in symbols $A_1 \multimap A_2$, if $h_1 \equiv \neg h_2$ (note that $\multimap$ is symmetric); finally, A_1 attacks A_2, in symbols $A_1 \leadsto A_2$, if (i) $A_1 \multimap A_2$ or $A_1 \rightsquigarrow A_2$ and, (ii) if $A_2 \multimap A_1$ or $A_2 \rightsquigarrow A_1$, $A_2 \not\succ A_1$.

The semantics of Dung's argumentation framework is based on the two notions of defence and conflict-freeness.

Definition 1. A set of arguments S defends an argument A iff, for each argument $B \in \mathtt{Arg}$ such that $B \leadsto A$, there exists an argument $C \in S$ such that $C \leadsto B$.

Definition 2. A set of arguments S is conflict-free iff there are no $A, B \in S$ such that $A \leadsto B$.

The following definition summarizes various semantics of acceptable arguments proposed in the literature. The output of the argumentation framework is derived from the set of acceptable arguments which are selected with respect to an acceptability semantics.

Definition 3. Let $S \subseteq \mathtt{Arg}$.

- S is *admissible* iff it is conflict-free and defends all its elements.
- A conflict-free S is a *complete extension* iff $S = \{A \mid S \text{ defends } A\}$.
- S is a *grounded extension* iff it is the smallest (for set inclusion) complete extension.
- S is a *preferred extension* iff it is a maximal (for set inclusion) complete extension.
- S is a *stable extension* iff it is a preferred extension that attacks all arguments in $\mathtt{Arg} \backslash S$.

In this paper we use the unique grounded extension, written as $E(\mathtt{Arg}, \succeq)$. Many properties and relations among these semantics have been studied by Dung and others.

Example 1. The example of Section 1 can be formalized as follows in terms of arguments.

$a = \langle \{bt \rightarrow wd, hi \rightarrow bt, hi\}, wd \rangle$, $b = \langle \{eg \rightarrow \neg hi, eg\}, \neg hi) \rangle$,
$c = \langle \{de \rightarrow \neg eg, de\}, \neg eg \rangle$, $c \succeq b$, $d = \langle \{hb \rightarrow wd, sm \rightarrow hb, sm\}, wd \rangle$,
$e = \langle \{rb \rightarrow \neg sm, rb\}, \neg sm \rangle$, $f = \langle \{ec \rightarrow \neg rb, ec\}, \neg rb \rangle$, $f \succeq e$,
$b \rightsquigarrow a$, $c \rightsquigarrow b$, $e \rightsquigarrow d$, $f \rightsquigarrow e$,
$\mathtt{Arg} = \{a, b, c, d, e, f\}$,
$E(\mathtt{Arg}, \succeq) = \{a, c, d, f\}$.

3 Arguments and Belief Revision

Belief revision is the process of changing beliefs to take into account a new piece of information. Traditionally the beliefs are modelled as propositions and the new piece of information is a proposition. In our model, instead, the belief base is made of arguments, and the new information is an argument too.

Let $*$ be an argumentative belief revision operator, it is defined as the addition of the new argument to the base as the one with the highest priority. Given $A = \langle H, h\rangle$, a base of arguments Q and a priority relation $\succeq_Q$ over Q:

$$\langle Q, \succeq_Q\rangle * A = \langle Q \cup \{A\}, \succeq_{(Q,\{A\})}\rangle \tag{1}$$

where $\succeq_Q \subset \succeq_{(Q,\{A\})} \wedge \forall A' \in Q\, A \succ_{(Q,\{A\})} A'$.

The new belief set can be derived from the new extension $E(Q \cup \{A\}, \succeq_{(Q,\{A\})})$ as the set of conclusions of arguments:

$$B(Q \cup \{A\}, \succeq_{(Q,\{A\})}) = \{h \mid \exists \langle H, h\rangle \in E(Q \cup \{A\}, \succeq_{(Q,\{A\})})\}. \tag{2}$$

Note that, given this definition, there is no warranty that the conclusion h of argument A is in the belief set; indeed, even if A is now the argument with highest priority, in the argument set Q there could be some argument A' such that $A' \rightleftharpoons_{\circ} A$. An argument $A' = \langle H', h'\rangle \multimap A$ (i.e., $h' \equiv \neg h$) would not be able to attack A, since $A \succ_Q A'$ by definition of revision. Instead, if $A' \rightsquigarrow A$, it is possible that A does not undercut or rebut A' in turn, and, thus, $A' \rightleftharpoons_{\circ} A$, possibly putting it outside the extension if no argument defends it against A'.

Success can be ensured only if the argument A can be supported by a set of arguments S with $\succeq_S$ which, once added to Q, can defend A in Q and defend themselves too.

Thus, it is necessary to extend the definition above to sets of arguments, to allow an argument to be defended:

$$\langle Q, \succeq_Q\rangle * \langle S, \succeq_S\rangle = \langle Q \cup S, \succeq_{(Q,S)}\rangle \tag{3}$$

where the relative priority among the arguments in S is preserved, and they have priority over the arguments in Q:

$\succeq_Q \subset \succeq_{(Q,S)} \wedge$
$\forall A', A'' \in S\, A' \succ_{(Q,S)} A''$ *iff* $A' \succ_S A'' \wedge$
$\forall A \in S, \forall A' \in Q\, A \succ_{(Q,S)} A'$.

Example 2. $Q = \{e\}, S = \{d, f\}, d \succ_S f, f \succ_S d$,
$\langle Q, \succeq_Q\rangle * S = \langle Q \cup S, \succeq_{(Q,S)}\rangle$,
$E(Q \cup S, \succeq_{(Q,S)}) = \{d, f\}$,
$d \succ_{(Q,S)} e, f \succ_{(Q,S)} e, d \succ_{(Q,S)} f, f \succ_{(Q,S)} d$,
$B(E(\{d, e, f\}, \succeq_{(Q,S)})) = \{wd, sm\}$.

4 An Abstract Agent Model

The basic components of our language are *beliefs* and *desires*. Beliefs are represented by means of an *argument base*. A belief set is a finite and consistent set of propositional formulas describing the information the agent has about the world and internal information. Desires are represented by means of a *desire set*. A desire set consists of a set of propositional formulas which represent the situations the agent would like to achieve. However, unlike the belief set, a desire set may be inconsistent, e.g., $\{p, \neg p\}$.

Let $\mathscr{L}$ be a propositional language.

Definition 4. The agent's desire set is a possibly inconsistent finite set of sentences denoted by D, with $D \subseteq \mathscr{L}$.

Goals, in contrast to desires, are represented by consistent desire sets.

We assume that an agent is equipped with two components:

- an argument base $\langle \mathtt{Arg}, \succeq_{\mathtt{Arg}} \rangle$ where $\mathtt{Arg}$ is a set of arguments and $\succeq_{\mathtt{Arg}}$ is a priority ordering on arguments.
- a desire set: $D \subseteq \mathscr{L}$;

The mental state of an agent is described by a pair $\Sigma = \langle \langle \mathtt{Arg}, \succeq_{\mathtt{Arg}} \rangle, D \rangle$. In addition, we assume that each agent is provided with a goal selection function G, and a belief revision operator $*$, as discussed below.

Definition 5. We define the belief set, B, of an agent, i.e., the set of all propositions in $\mathscr{L}$ the agent believes, in terms of the extension of its argument base $\langle \mathtt{Arg}, \succeq_{\mathtt{Arg}} \rangle$:

$$B = B(\mathtt{Arg}, \succeq_{\mathtt{Arg}}) = \{h \mid \exists \langle H, h \rangle \in E(\mathtt{Arg}, \succeq_{\mathtt{Arg}})\}.$$

We will denote by $\Sigma_{\mathscr{S}}$, $\mathtt{Arg}_{\mathscr{S}}$, $E(\mathtt{Arg}_{\mathscr{S}}, \succeq_{\mathtt{Arg}})$ and $B_{\mathscr{S}}$, respectively, the mental state, the argument base, the extension of $\mathtt{Arg}_S$, and the belief set of an agent $\mathscr{S}$.

In general, given a problem, not all goals are *achievable*, i.e. it is not always possible to construct a plan for each goal. The goals which are not achievable or those which are not chosen to be achieved are called *violated goals*. Hence, we assume a problem-dependent function $\mathscr{V}$ that, given a belief base B and a goal set $D' \subseteq D$, returns a set of couples $\langle D^a, D^v \rangle$, where D^a is a maximal subset of achievable goals and D^v is the subset of violated goals and is such that $D^v = D' \setminus D^a$. Intuitively, by considering violated goals we can take into account, when comparing candidate goal sets, what we lose from not achieving goals.

In order to act an agent has to take a decision among the different sets of goals he can achieve.

The aim of this section is to illustrate a qualitative method for goal comparison in the agent theory. More precisely, we define a qualitative way in which an agent can choose among different sets of candidate goals. Indeed, from a desire set D, several candidate goal sets D_i, $1 \leq i \leq n$, may be derived. How can an agent choose

among all the possible D_i? It is unrealistic to assume that all goals have the same priority. We use the notion of preference (or urgence) of desires to represent how relevant each goal should be for the agent depending, for instance, on the reward for achieving it. The idea is that an agent should choose a set of candidate goals which contains the greatest number of achievable goals (or the least number of violated goals).

We assume we dispose of a total pre-order $\succeq$ over an agent's desires, where $\phi \succeq \psi$ means desire ϕ is at least as preferred as desire ψ.

The $\succeq$ relation can be extended from goals to sets of goals. We have that a goal set D_1 is preferred to another one D_2 if, considering only the goals occurring in either set, the most preferred goals are in D_1. Note that $\succeq$ is connected and therefore a total pre-order, i.e., we always have $D_1 \succeq D_2$ or $D_2 \succeq D_1$ (or both).

Definition 6. Goal set D_1 is at least as important as goal set D_2, denoted $D_1 \succeq D_2$ iff the list of desires in D_1 sorted by decreasing preference is lexicographically greater than the list of desires in D_2 sorted by decreasing importance. If $D_1 \succeq D_2$ and $D_2 \succeq D_1$, D_1 and D_2 are said to be indifferent, denoted $D_1 \sim D_2$.

However, we also need to be able to compare the mutual exclusive subsets (achievable and violated goals) of the considered candidate goal, as defined below.

We propose two methods to compare couples of goal sets.

Given the $\succeq_D$ criterion, a couple of goal sets $\langle D_1^a, D_1^v \rangle$ is at least as preferred as the couple $\langle D_2^a, D_2^v \rangle$, noted $\langle D_1^a, D_1^v \rangle \succeq_D \langle D_2^a, D_2^v \rangle$ iff $D_1^a \succeq D_2^a$ and $D_1^v \preceq D_2^v$.
$\succeq_D$ is reflexive and transitive but partial. $\langle D_1^a, D_1^v \rangle$ is strictly preferred to $\langle D_2^a, D_2^v \rangle$ in two cases:

1. $D_1^a \succeq D_2^a$ and $D_1^v \prec D_2^v$, or
2. $D_1^a \succ D_2^a$ and $D_1^v \preceq D_2^v$.

They are indifferent when $D_1^a = D_2^a$ and $D_1^v = D_2^v$. In all the other cases, they are not comparable.

Given the $\succeq_{Lex}$ criterion, a couple of goal sets $\langle D_1^a, D_1^v \rangle$ is at least as preferred as the couple $\langle D_2^a, D_2^v \rangle$ (noted $\langle D_1^a, D_1^v \rangle \succeq_{Lex} \langle D_2^a, D_2^v \rangle$) iff $D_1^a \sim D_2^a$ and $D_1^v \sim D_2^v$; or there exists a $\phi \in \mathscr{L}$ such that both the following conditions hold:

1. $\forall \phi' \succeq \phi$, the two couples are indifferent, i.e., one of the following possibilities holds: (a) $\phi' \in D_1^a \cap D_2^a$; (b) $\phi' \notin D_1^a \cup D_1^v$ and $\phi' \notin D_2^a \cup D_2^v$; (c) $\phi' \in D_1^v \cap D_2^v$.
2. Either $\phi \in D_1^a \setminus D_2^a$ or $\phi \in D_2^v \setminus D_1^v$.

$\succeq_{Lex}$ is reflexive, transitive, and total.

In general, given a set of desires D, there may be many possible candidate goal sets. An agent in state $\Sigma = \langle \texttt{Arg}, D \rangle$ must select precisely one of the most preferred couples of achievable and violated goals.

Let us call G the function which maps a state Σ into the couple $\langle D^a, D^v \rangle$ of goal sets selected by an agent in state Σ. G is such that, $\forall \Sigma$, if $\langle \bar{D}^a, \bar{D}^v \rangle$ is a couple of goal sets, then $G(\Sigma) \succeq \langle \bar{D}^a, \bar{D}^v \rangle$, i.e., a rational agent always selects one of the most preferable couple of candidate goal sets [3].

5 An Abstract Model of Speaker-Receiver Interaction

Using the above agent model, we consider two agents, $\mathscr{S}$, the speaker, and $\mathscr{R}$, the receiver. $\mathscr{S}$ wants to convince $\mathscr{R}$ of some proposition p.

How does agent $\mathscr{S}$ construct a set of arguments S? Of course, $\mathscr{S}$ could include all the arguments in its base, but in this case it would risk to make his argumentation less appealing and thus to make $\mathscr{R}$ refuse to revise its beliefs, as discussed in the next section. Thus, we require that the set of arguments S to be communicated to $\mathscr{R}$ is minimal: even if there are alternative arguments for p, only one is included.

We include the requirement that S is chosen using arguments which are not already believed by $\mathscr{R}$. S is a minimal set among the T defined in the following way:

$$T \subseteq \mathtt{Arg}_{\mathscr{S}} \wedge B(\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}} \rangle * \langle T, \succeq_{\mathscr{S}} \rangle)) \vdash p. \tag{4}$$

Example 3. $S = \{a,c\}, p = wd, \mathtt{Arg}_{\mathscr{R}} = \{b\} E(\mathtt{Arg}_{\mathscr{R}} \cup S, \succeq_{(\mathtt{Arg}_{\mathscr{R}},S)}) = \{a,c\}$, $B(E(\mathtt{Arg}_{\mathscr{R}} \cup S, \succeq_{(\mathtt{Arg}_{\mathscr{R}},S)})) = \{wd, \neg eg\}$.

This definition has two shortcomings: first, such an S may not exist, since T could be empty. There is no reasonable way of assuring that $\mathscr{S}$ can always convince $\mathscr{R}$: as we discussed in Section 3, no success can be assumed.

Second, in some cases arguments in $E(\mathtt{Arg}_{\mathscr{R}} \cup S, \succeq_{(\mathtt{Arg}_{\mathscr{R}},S)})$ may be among the ones believed by $\mathscr{R}$ but not by $\mathscr{S}$. If they contribute to prove p, there would be a problem: $\exists A \in \mathtt{Arg}_{\mathscr{R}} \backslash \mathtt{Arg}_{\mathscr{S}}\, B(E((\mathtt{Arg}_{\mathscr{R}} \backslash \{A\}) \cup S, \succeq_{(\mathtt{Arg}_{\mathscr{R}},S)})) \nvdash p$

This would qualify $\mathscr{S}$ as a not entirely sincere agent, since he would rely (even if he does not communicate them explicitly) on some arguments he does not believe, which are used in the construction of the extension from which p is proved.

The second problem, instead, can be solved in the following way, by restricting set S not to require arguments not believed by $\mathscr{S}$ to defend S. S is now a minimal T such that $T \subseteq \mathtt{Arg}_{\mathscr{S}}$ and $B(\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}} \rangle * \langle T, \succeq_{\mathscr{S}} \rangle) \vdash p$ and $\neg \exists A \in \mathtt{Arg}_{\mathscr{R}} \backslash \mathtt{Arg}_{\mathscr{S}}\, B(\langle \mathtt{Arg}_{\mathscr{R}} \backslash \{A\}, \succeq_{\mathtt{Arg}_{\mathscr{R}}} \rangle * \langle T, \succeq_{\mathscr{S}} \rangle) \nvdash p$

Example 4. $\mathtt{Arg}_{\mathscr{S}} = \{a,c,i\}, \mathtt{Arg}_{\mathscr{R}} = \{b,g,h\}, g \rightsquigarrow c, h \rightleftharpoons g, i \rightleftharpoons g.$
If $S = \{a,c\}, p = wd$: $E(\mathtt{Arg}_{\mathscr{R}} \cup S, \succeq_{(\mathtt{Arg}_{\mathscr{R}},S)}) = \{a,c,h\}$,
$B(\{a,b,c,g,h\}) = \{wd, \neg eg \ldots\}$.
If $S = \{a,c,i\}, p = wd$: $E(\mathtt{Arg}_{\mathscr{R}} \cup S, \succeq_{(\mathtt{Arg}_{\mathscr{R}},S)}) = \{a,c,i\}$,
$B(\{a,b,c,g,h,i\}) = \{wd, \neg eg \ldots\}$.

The belief revision system based on argumentation (see Section 2), is used to revise the public face of agents: the agents want to appear rational (otherwise they lose their status, reliability, trust, etc.) and, thus, when facing an acceptable argument (i.e., they do not know what to reply) have to admit that they believe it and to revise the beliefs which are inconsistent with it.

We want to model social interactions among agent which do not necessarily tell the truth or trust each other completely, although they may pretend to. In such a setting, an agent revises its private beliefs only if someone provides an acceptable argument in the sense of Section 2.

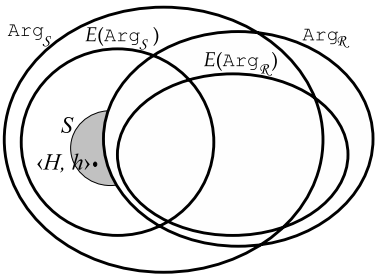


Fig. 1 A diagram of mutual inclusion relations among the belief bases and sets involved in the interaction between $\mathscr{S}$ and $\mathscr{R}$.

Thus, while publicly an agent must pretend to be rational and thus shall revise its public belief base according to the system discussed in Section 3, nothing forbids an agent to privately follow other types of rules, not even necessarily rational. As a worst-case scenario (from $\mathscr{S}$'s standpoint), we assume that $\mathscr{R}$ uses a belief revision system based on Galbraith's notion of conventional wisdom discussed in [2] as a proposal to model the way an irrational (but realistic) agent might revise its private beliefs.

The idea is that different sets of arguments $S_1, \ldots, S_n$ lead to different belief revisions $\langle \mathtt{Arg}, \succeq_{\mathtt{Arg}} \rangle * \langle S_1, \succeq_{S_1} \rangle, \ldots, \langle \mathtt{Arg}_{\mathtt{Arg}}, \succeq_{\mathtt{Arg}} \rangle * \langle S_n, \succeq_{S_n} \rangle$. $\mathscr{R}$ will privately accept the most appealing argument, i.e., the S_i which maximizes the preferences according to the notion of Galbraith's conventional wisdom.

In order to formalize this idea, we have to define an order of *appeal* on sets of beliefs.

Definition 7. Let $\mathtt{Arg}_1$ and $\mathtt{Arg}_2$ be two argument bases. $\mathtt{Arg}_1$ is *more appealing* than $\mathtt{Arg}_2$ to an agent, with respect to the agent's desire set D, in symbols $\mathtt{Arg}_1 \succeq \mathtt{Arg}_2$, if and only if $G(\langle \langle \mathtt{Arg}_1, \succeq_{\mathtt{Arg}_1} \rangle, D \rangle) \succeq G(\langle \langle \mathtt{Arg}_2, \succeq_{\mathtt{Arg}_2} \rangle, D \rangle)$.

We will denote by • the private, CW-based belief revision operator. Given an acceptable argument set S,

$$\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}} \rangle \bullet \langle S, \succeq_S \rangle \in \{ \langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}} \rangle, \langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}} \rangle * \langle S, \succeq_S \rangle \}.$$

This definition is inspired to indeterministic belief revision [6]: "Most models of belief change are deterministic. Clearly, this is not a realistic feature, but it makes the models much simpler and easier to handle, not least from a computational point of view. In indeterministic belief change, the subjection of a specified belief base to a specified input has more than one admissible outcome.

Indeterministic operators can be constructed as sets of deterministic operations. Hence, given n deterministic revision operators $*_1, *_2, \ldots, *_n$, $* = \{*_1, *_2, \ldots, *_n\}$ can be used as an indeterministic operator."

We then define the notion of appealing argument, i.e., an argument which is preferred by the receiver $\mathscr{R}$ to the current state of its beliefs:

Definition 8. Let S be a minimal set of arguments that supports $A = \langle H, p\rangle$, such that S defends A and defends itself, as defined in the previous section:
$\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle \bullet \langle S, \succeq_S\rangle = \langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S, \succeq_S\rangle$,
i.e., $\mathscr{R}$ privately accepts revision $\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S, \succeq_S\rangle$, if

$$\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S, \succeq_S\rangle \succeq \langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle$$

otherwise $\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle \bullet \langle S, \succeq_S\rangle = \langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle$.

Example 5. The investor of our example desires investing money. Assuming this is his only desire, we have $D_{\mathscr{R}} = \{im\}$. Now, the advisor $\mathscr{S}$ has two sets of arguments to persuade $\mathscr{R}$ that the dollar is weak, namely $S_1 = \{a, c\}$ and $S_2 = \{d, f\}$. Let us assume that, according to the "planning module" of $\mathscr{R}$,

$$\mathscr{V}(\langle\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S_1, \succeq_{S_1}\rangle, D_{\mathscr{R}}\rangle) = \langle\{im\}, \emptyset\rangle,$$
$$\mathscr{V}(\langle\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S_2, \succeq_{S_2}\rangle, D_{\mathscr{R}}\rangle) = \langle\emptyset, \{im\}\rangle.$$

Therefore, $G(\langle\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S_1, \succeq_{S_1}\rangle, D_{\mathscr{R}}\rangle) \succeq G(\langle\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle * \langle S_2, \succeq_{S_2}\rangle, D_{\mathscr{R}}\rangle)$, because, by revising with $S_1 = \{a, c\}$, $\mathscr{R}$'s desire *im* is achievable.

A necessary and sufficient condition for the public and private revisions to coincide is thus that the set of arguments S used to persuade an agent is the most appealing for the addressee, if one exists.

Since CW-based belief revision is indeterministic and not revising is an alternative, $\mathscr{R}$ decides whether to keep the *status quo* of his beliefs or to adopt the belief revision resulting from the arguments proposed by $\mathscr{S}$.

Seen from $\mathscr{S}$'s standpoint, the task of persuading $\mathscr{R}$ of p is about comparing $\mathscr{R}$'s belief revisions resulting from the different sets of arguments supporting p and acceptable by $\mathscr{R}$, and choosing the set of arguments that appeals most to $\mathscr{R}$.

To define the notion of the most appealing set of arguments, we need to extend the order of appeal $\succeq$ to sets of arguments.

Definition 9. Let S_1 and S_2 be two sets of arguments that defend themselves; S_1 is more appealing to $\mathscr{R}$ than S_2, in symbols $S_1 \succeq_{\mathscr{R}} S_2$, if and only if

$$\langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle \bullet \langle S_1, \succeq_{S_1}\rangle \succeq \langle \mathtt{Arg}_{\mathscr{R}}, \succeq_{\mathtt{Arg}_{\mathscr{R}}}\rangle \bullet \langle S_2, \succeq_{S_2}\rangle.$$

The most appealing set of arguments S_p^* for persuading $\mathscr{R}$ of p, according to conventional wisdom, is, among all minimal sets of arguments S that support an $A = \langle H, p\rangle$, such that S defends A and S defends itself as defined in Section 5, the one that is maximal with respect to the appeal $\succeq_{\mathscr{R}}$, i.e., such that $S_p^* \succeq_{\mathscr{R}} S$.

6 Conclusions

We studied how to choose arguments in persuasion to maximize their acceptability with respect to their receiver. In some applications, when agents have to interact with

human users who act in a non fully rational way, like, e.g., following the principle of conventional wisdom, it is necessary to model such a behavior.

To model the process of selecting acceptable arguments, in this paper:

- We derive the beliefs of an agent from a base of arguments. An agent believes the propositions which are supported by the arguments of the grounded extension of its argument base.
- We propose a definition of belief revision of an argument base as an expansion of the base with the new arguments and by giving priority to the last introduced argument.
- We define the notion of appeal of an argument in terms of the goals which the revision triggered by the argument allows to satisfy by means of a plan.

It would be interesting to investigate how the work by Hunter [7] relates with conventional wisdom and our definition of appeal. Note that appeal must not be confused with wishful thinking: the receiver does not prefer a state of the world which makes its goals true, but one which gives him more opportunities to act to achieve its goals. The rationality of this kind of reasoning is discussed, e.g., by [1].

In this paper we do not study the formal properties of argumentative belief revision, and we do not relate it to the AGM postulates. However, from the paper it already appears that postulates like success are not meaningful in this framework. Moreover, we do not study how the different types of argumentation frameworks impact on belief revision [9].

References

1. G. Boella, C. da Costa Pereira, A. Tettamanzi, G. Pigozzi, and L. van del Torre. What you should believe: Obligations and beliefs. In *Proceedings of KI07-Workshop on Dynamics of Knowledge and Belief*, 2007.
2. G. Boella, C. D. C. Pereira, G. Pigozzi, A. Tettamanzi, and L. van der Torre. Choosing your beliefs. In G. Boella, L. W. N. van der Torre, and H. Verhagen, editors, *Normative Multi-agent Systems*, volume 07122 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2007.
3. C. da Costa Pereira and A. Tettamanzi. Towards a framework for goal revision. In *BNAIC 2006*, pages 99–106. University of Namur, 2006.
4. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and *n*-person games. *Artificial Intelligence*, 77(2):321–358, 1995.
5. J. K. Galbraith. *The Affluent Society*. Houghton Mifflin, Boston, 1958.
6. S. O. Hansson. Logic of belief revision. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Summer 2006.
7. A. Hunter. Making argumentation more believable. In *AAAI 2004*, pages 269–274, 2004.
8. F. Paglieri and C. Castelfranchi. Revising beliefs through arguments: Bridging the gap between argumentation and belief revision in MAS. In I. Rahwan, P. Moraitis, and C. Reed, editors, *ArgMAS 2004*, volume 3366 of *Lecture Notes in Computer Science*, pages 78–94. Springer, 2005.
9. N. D. Rotstein, A. J. Garcia, and G. R. Simari. From desires to intentions through dialectical analysis. In *AAMAS '07*, pages 1–3, New York, NY, USA, 2007. ACM.

Foundation for Virtual Experiments to Evaluate Thermal Conductivity of Semi- and Super-Conducting Materials

R. M. Bhatt

Department of Computer Science

HNB Garhwal University, Srinagar Garhwal 246 174 India
rmbhatt77@yahoo.com

R. P. Gairola

Department of Physics, Birla Campus

HNB Garhwal University, Srinagar Garhwal 246 174

Abstract Thermal conductivity of solids provides an ideal system for analysis by conducting numerical experiment, currently known as virtual experiment. Here, the model is a numerical model, which is dynamic in nature, as the parameters are interrelated. The present paper discusses the steps involved to conduct virtual experiments using Automated Reasoning for simulation to evaluate the thermal conductivity of Ge, Mg_2Sn semiconducting and YBCO superconducting materials, close to the experimental values.

1. Introduction

Computers can help human to be creative[1] in a number of ways e.g. providing a continuous interaction between the man and machine, requires an even deeper understanding of the subject concerned. AI techniques are required to put the efforts near to the actual experiment in most economical manner. However, its considerable applications have not been applied in the thermal science [12]. To execute the Virtual Experiment (VE), considering various parameters, a model has been designed. Using Automated Reasoning(AR) for simulation, we find the fitness proven to be a reasonable facsimile of real experimental values for the thermal conductivity of Germanium (Ge), Magnesium stannide (Mg_2Sn) semi-conducting and Yttrium Barium Cupric Oxide (YBCO) superconducting materials.

Please use the following format when citing this chapter:

Bhatt, R.M. and Gairola, R.P., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 225234.

2. Foundation for Virtual Experiment

Belonging to different fields viz. Economics, Physics, Biology etc., simulated approach has been applied e.g. usefulness of simulation, using models of economic systems [13], is reported as "Simulation can be used to experiment with new situations about which little or no information is available, so as to prepare for what may happen". Which is also described [19] as the process of designing a computerized model to conduct experiments. This examines the nature of human intelligence by doing soft computing that mimic the intelligence behaviour [6]. In AR, programs are written to prove mathematical theorems and it has been used as a reasoning engine to discover the knowledge. Here, the propositional logic and an alternative representation for proposition clauses have been used.

2.1 – Applications of simulation

Reasons can be given in favour of the VE as [17] " Such refinements provide a better understanding of physical problems which can not be obtained from experiment". AI techniques are used in physical science e.g. phase transformation [14]; for predictions [16]; to identify the distillation process [18]; and to design the complex thermal system [15].
Due to their inherent peculiar properties, semiconducting and superconducting systems promise wide applications. The various models, needed to solve a complex problem are mentioned in Fig. 2.1.

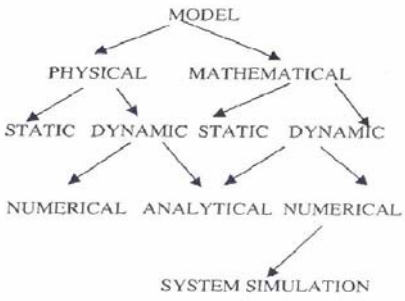


Fig.-1 : Types of Models

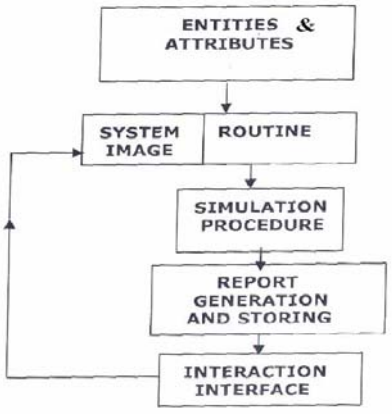


Fig.-2 :Flow Chart for Controlled Simulation

2.1.1 Stages for Simulation Task

There are mainly five stages mapped-out for preparing for simulation as shown in the Fig. 2.1.1. An additional stage of interaction interface is being considered in the earlier 4 stages for simulation task[11]. This modification helps in controlling the simulation process. The first stage lists all parameters and activities. The second stage is to design the model by fitting the parameters and activities into the system image and routines separately to act like a model collectively. Thirdly, simulation algorithm is defined depending upon behaviour of the parameters. In the fourth stage, simulated responses are generated. In the fifth stage, interaction parameters are defined to provide a kind of feed back and help to retain the state of simulation and doing the repetitive process as required.

2.1.2 Automated Reasoning

Arithmetic and logical conditions have been applied and manipulated to decide whether the simulated results be accepted or neglected. The general format of the alternative representation for the propositional clause applied is:

IF < > THEN < > ELSE < >

During the simulation process, these conditions are applied and tested to get the best possible theoretical observations for fitting with the experimental values. The set of conditions as defined above are tested by using the logical AND operator. To gather the knowledge and to infer the simulated response for the fitness, rule-based systems has been applied as shown in the Fig. 2.1.2 of the logic tree.

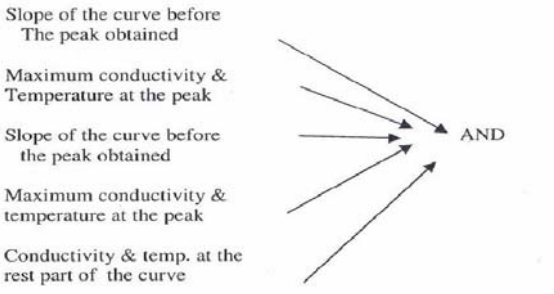


Fig. 2.1.2 : Logic Tree

2.2 - Applications of two Dimensional Arrays

By providing the feed back, interactively, appropriate values of different parameters are processed for the fitness of the hypothesis as shown in the Fig. 2.2, the

self-explanatory diagram describing the interfacing algorithm. The preliminary development of this approach has been partially reported elsewhere[3].
The values for the conductivity (K) have been generated in the form of a 2-D matrix/arrays as RESS(I,J), for a set of parameters, while the value of one of them has been altered. For a set of constant values of parameters - A, γ, α, β and, $\in$ and altered parameter δ, a 2-D array of temperature v/s δ is shown in Table 2.2.

Table 2.2 : Storage of responses

Temp/δ	200	210	220	230
50	4.92	4.77	4.64	4.51
100	4.30	4.15	4.00	3.86
120	3.94	3.79	3.65	3.52
160	3.29	3.15	3.02	2.91

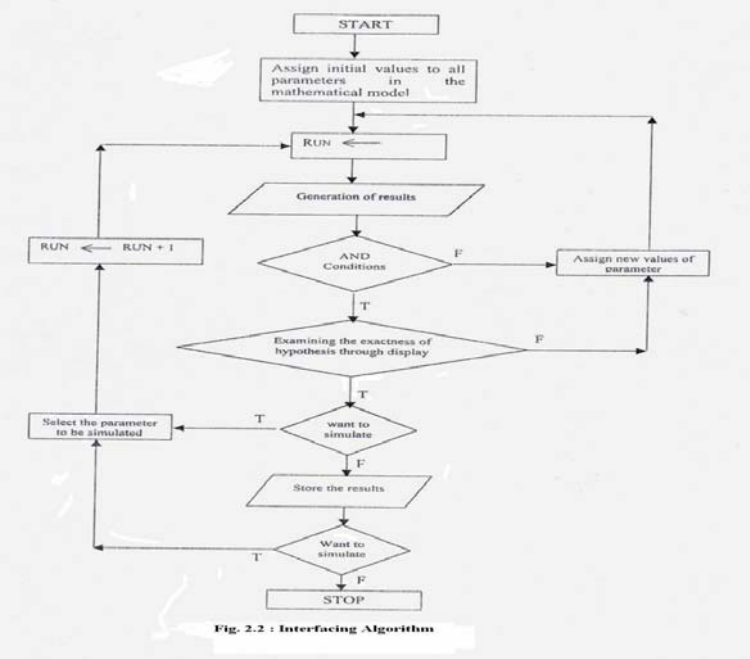


Fig. 2.2 : Interfacing Algorithm

3. Mathematical Model and Virtual Experiment

The problem of integral calculations occurs very often in thermal science, where many parameters are involved to understand the nature of various scattering proc-

esses, operating at different temperatures simultaneously. To evaluate the thermal conductivity, theoretical model (numerical) of Callaway[7] has been considered.

3.1 Rule for Numerical Integration

The functions in the theoretical physics besides being continuous are usually one or more times differentiable. Therefore, to increase the accuracy of the numerical integration for an equal number of mesh points, the Simpson rule[8] is applied. The mesh width of each interval between a and b, can be defined as, h =(b-a)/n, n(even) is the sub-subintervals where a=.00001, b=20.0 and n=100 have been taken into account. The error is only of the order of h^4 , so precision is under controlled.

3.2 Algorithm For Virtual Experimentation

Logic is developed to execute the desired work and a computer program is developed accordingly as shown in the Interfacing Algorithm diagram. To compute speedily and to overcome the repetitive programming steps, subroutines are preferred. DO statement is extensively applied for various reasons, especially for arrays and subroutine handling. The computer program is developed in the FORTRAN-77 language[9].

4. Test for different cases to evaluate thermal conductivity

We have executed the above discussed logic on the proposed model, for instance, for the Ge, a semi-conducting material. After successfully testing its conductivity results in the temperature range from 2^o K to 10^o K, we have proceeded further for detailed computations for the conductivity analyses for Ge & Mg_2Sn semiconducting and YBCO superconducting samples.

4.1 Test for Germanium(Ge) Semiconductor

In analyzing the phonon conductivity of germanium, following equation for the thermal induced phonon relaxation rate is required,

$$\tau^{-1} = v/FL + A\,\omega^4 + (B_1+B_2)\,\omega^2\,T^3 + D\,\omega^3\,T \qquad (4.1)$$

Here v is the sound velocity, T is the temperature and other symbols are the various parameters needed to test a particular theory. Values of different parameters used in the calculation for a preliminary test are taken from the earlier work[10], wherein the use of a computer program for achieving fitness in the wide range of temperature has been insisted. The set of values are :

$v=3.5\times10^{5}$ cm/s; L=.24 cm' F=.8; $\theta_D=376$; $A=2.4\times10^{-44}$ s^5

$B_1+B_2=2.77\times10^{-33}$ sec K^{-3}; $D=1.203\times10^{-33}$ s^2 K^{-1}.

Test shows accuracy with the experimental results for the temperature 2 ^oK, 4 ^oK, 6 ^oK, and 10 ^oK, which are .474 , .261 , .504 , .791 and .985 $W/cm^{-1}K^{-1}$, respectively. Due to fitness of test, it is further carried up to the temperature of 40^o K. The Table 4.1.1 illustrates the different values of the parameters and their simulated inferences for the conductivity values are shown in the Table 4.1.2.

Table 4.1.1: Parameters and Values for Ge

Parameters ↓	Values →			
	I	II	III	IV
v(x10^5 cm/sec.)	3.5	3.5	3.5	3.5
L	.24	.24	.24	.243
F	.80	.77	.77	.77
θ_D	376	376	376	376
A (x 10^{-44} $sec.^3$)	2.4	2.4	2.4	2.4
B_1+B_2 (x 10^{-23} $sec.K^{-3}$)	2.77	3.43	3.43	3.43
D (x 10^{-33} $sec.^3K^{-1}$)	1.203	1.433	3.423	3.334
Max. Conductivity(x10^7)	21.50	18.61	12.62	12.83
(at Temp.^oK)	17	16	18	18

Table 4.1.2: Thermal Conductivity measures for Ge

Re-sponse	K (x10^7) ↓	Temp 2 →	4	8	10	15	20	25	30	35	40
I	Cond.	.49	3.17	1.20	1.58	2.10	2.10	1.26	1.58	1.32	1.10
II	Cond.	.47	2.98	1.09	1.42	1.80	1.80	1.52	1.32	1.05	0.91
III	Cond.	.45	2.52	7.79	0.97	1.20	1.24	1.09	0.97	0.80	0.70
IV	Cond.	.46	2.56	7.93	0.99	1.22	1.26	1.11	0.98	0.81	0.71

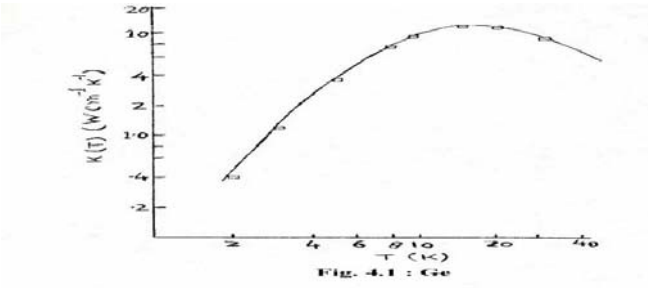


Fig. 4.1 : Ge

These four inferences for thermal conductivity measurements are closely examined and the values shown against the IVth observations (marked with *) are found fit, graphically depicted in the Fig. 4.1, where circle shows the experimental point and the present analysis has been shown as the curve-line.

4.2 Test for Magnesium Stannide (Mg_2Sn) Semiconductor

We consider the following expression for relaxation time,

$$\tau^{-1}(\omega) = (v/FL)+A\omega^4+[B_1+B_2\exp.(-\Theta/aT)]\omega^2T^3+D\omega^3T \qquad (4.2)$$

It has been a usual practice, to generally neglect the exponential temperature dependence of the parameter B_2 for the conductivity calculation, representing Umklapp phonon-scattering and both B_1 (normal phonon scattering parameter) and $B_2$are lumped into a single parameter B, assumed to be independent of T. Therefore, B_2 is taken to depend upon T, exponentially, in the analysis. Table 4.2.2 shows four simulated response against the values of different parameters as shown in the Table 4.2.1.

Table 4.2.1: Parameters and Values for Mg_2Sn

↓ Parameters	Values →			
	I	II	III	IV
v(x10^5 cm/sec.)	359	359	359	359
L	.11	.11	.10	.10
F	.54	.54	.54	.54
$\propto$	2.0	2.5	2.5	2.5
θ_D	154	154	154	154
A (x 10^{-44} sec.^{3})	6.3	6.3	6.3	6.3
B_1(x 10^{-23}sec.K^{-3})	7.0	7.0	7.7	7.7
B_2(x 10^{-23}sec.K^{-3})	4.7	4.7	4.7	4.7
D (x 10^{-33}sec.^{3}K^{-1})	2.75	2.75	2.75	2.95
Max. Conductivity(x10^7)	6.483	6.398	6.088	5.909
(at Temp.^{o}K)	16	14	14	14

Table 4.2.2: Thermal Conductivity measures for Mg_2Sn

Re-sponse	K ↓ (x10^7)	Temp 2 →	6	8	10	14	20	26	30	36	40
I	Cond.	.96	2.37	3.89	5.16	6.45	5.74	4.08	3.15	2.15	1.69
II	Cond.	.96	2.37	3.89	5.16	6.39	5.47	3.71	2.79	1.87	1.46*
III	Cond.	.89	2.21	3.65	4.87	6.08	5.25	3.57	2.70	1.81	1.42
IV	Cond.	.68	2.17	3.57	4.74	5.90	5.10	3.49	2.64	1.78	1.40

The corresponding results for thermal conductivity are examined and the values of the II observations(marked with *) are found fit, shown by the curve-line in Fig.4.2 where the experimental data is shown as circle-points.

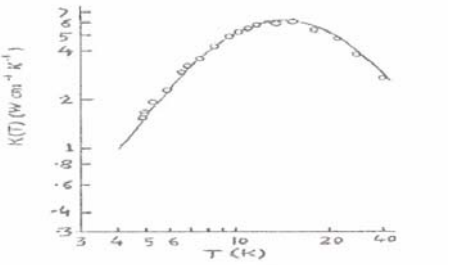


Fig. 4.2 : Mg2Sn

4.3 Test for Yttrium Barium Cupric Oxide (YBCO) superconductors

In computing the thermal conductivity of YBCO superconductors, behaviour has also been examined by earlier workers [2]. We have considered the Callaway's model, which is also used by Tewordt et al.[20] in a modified form-

$$K = A\, t^3 \int x4\, e^x / [(e^x - 1)^2 \,.\, F(t, x)]\, dx \qquad (4.3.1)$$

$$F(t,x)=[1+\alpha x^4t^4+\beta x^2t^2+ \gamma txg(x,y)+ \delta x^3t^4+(\in x^2t^5] \qquad (4.3.2)$$

A, α, β, γ, δ and $\in$ are scattering strengths due to boundary scattering, point defect scattering, sheet like fault, electron-phonon scattering, interference scattering and three phonon scattering. Corresponding maximum conductivity values are shown in the Table 4.3.1.

Table 4.3.1: Thermal Conductivity measures for YBCO

Response	Max.Cond.	Temp	A	α	β	γ	δ	$\in$
I	3.50	70	4	15	50	50	210	.01
II	3.82	60	4	25	50	50	210	.01
III	4.14	60	4	15	50	50	210	.01
IV	3.81	70	5	15	50	50	210	.01

Table 4.3.2: Thermal Conductivity measures for YBCO

Re-sponse	K ($\downarrow$ ($x10^7$))	Temp $\longrightarrow$								
		10	20	30	40	80	100	120	140	160
I	Cond.	.68	1.67	2.35	2.73	3.03	2.95	2.82	2.68	2.54
II	Cond.	1.39	2.73	3.40	3.70	3.70	3.49	3.24	2.99	2.75
III	Cond.	1.43	2.87	3.63	3.98	4.03	3.80	3.54	3.27	3.02
IV	Cond.	.86	2.09	2.93	3.41	3.79	3.68	3.53	3.35	3.17

We have found positive results in the temperature range from 10-160 ^{o}K and fitness (shown as curve-line) with the experimental results (shown as circle-point) from the IV observations of Table 4.3.2, as shown in the Fig. 4.3.

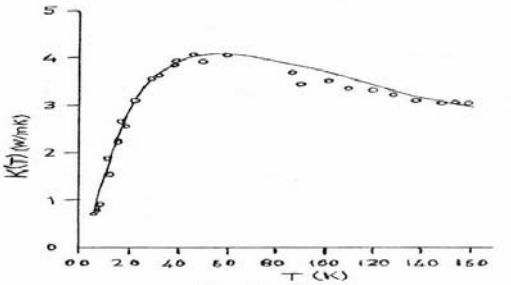


Fig. 4.3 : YBCO

5. Model Validation

The model has also been validated in two cases. First case of the semiconducting material Ge, shows[5] a good agreement between theory and experiment in the temperature range 2 to 100 ^{o}K. For the second case, similar approach also enables to analyse the three different samples of YBCO superconductors[4] in the temperature range 0 to 260 ^{o}K and the interference scattering & exponential temperature dependence lead to a good agreement with the experimental data.

6. CONCLUSION

It emerges that the VE has immense capabilities to yield good results, within the prescribed automated reasoning and the interface algorithm,. In performing VE over the different models for these materials (Ge, Mg and YBCO) , the various parameters have been considered so as to search for the unusual features or properties might provide a background for understanding the mechanisms.

7. ACKNOWLEDGEMNT

Dr. Bhatt wants to sincerely acknowledge the encouragement received from his respected and beloved father Sri Rameshwar Prasad Bhatt who passed away on 16 th of Feb., 2007. Computational facilities availed at Deptt. of Computer Science, HNB Garhwal University is also sincerely acknowledged.

References

1. Adkins Gerald and Pooch Udo W (1987) Computer Simulation: A tutorial, in selected reports in Software. In: Marvin V. Zelkowitz(ed) The Computer Society of the IEEE, 384:381-393
2. Aubin H Behnia K Ribault M Taillfer L and Gagnon R (1997) Zeit. fur Physik B-103:149
3. Bhatt R M and Gairola R P (1997) Simulation for conductivity curve-fitting in YBCO superconductors, In: Venkata Rao and T.P.Rama Rao (ed) IT for Organisational Excellence, TMH, New Delhi
4. Bhatt R M Gairola R P (2001) Simulation Analysis for lattice thermal conductivity of YBCO superconductors. Current Science 80(7):864-867
5. Bhatt R M Gairola R P (2002) Simulation Analysis for phonon conductivity of Germanium. In: proc. of XXXVII Annual CSI, Harnessing and managing knowledge, CD produced by www.productreach.net, Bangalore
6. Bonnet A (1985) A I, Promise and Performance. Prentice Hall, N.J.
7. Callaway Joseph (1959) Phys. Rev. 113: 1046.
8. Chandra Suresh (1995) FORTRAN programming and numerical techniques. Sultan chand & Sons, New Delhi
9. Digital Research Fortran Language reference manual (1983), first edition, release note 02-1985, California, USA
10. Gairola R P(1985) Phys. St. Sol. (b) 125: 65
11. Geoffrey Gordan (2007) System simulation, 2nd edn. Prentice-Hall of India, New Delhi
12. Krieth F Timmerhaus K Lior N Shaw H Shah R K Bell K J Diller K R Valvano J W (2000)Applications. In: Frank Kreith(ed) The CRC Hndbook of Thermal Engineering, Boca Raton: CRC Press LLC
13. Naylor T H (1971) Computer simulation experimentation with models of economic systems. John Wiley and sons, New York
14. Osetsky Yu N (1998) Computer-simulation study of high temperature phase stability in iron. Phys. Rev. B-57(2):755-763
15. Paoletti B Sciubba E (1997) Artificial Intelligence in thermal systems design: Concept and Applications. In: R. F. Boehm(ed) Developments in the design of thermal systems, Cambridge University Press, New York
16. Pavlovic A S Suresh Babu V and Mohindar Seehra S (1996) High-temperature thermal expansion of binary alloys of Ni with Cr, Mo and Re: a comparison with molecular dynamics simulation. J. Phys. Cond. Matter 8:3139-3144
17. Rajaraman V (1997) Simulation with supercomputers, in proc. of Int'l conf. on cognitive systems, Vol. II, Allied, New Delhi
18. Rasmussen K H Nielsen C S Jorgensen S (1990) Identification of distillation process dynamics comparing process knowledge and black box based approaches. In: Proced. Am. Control Conf. 3116-3121
19. Shanon R F (19975) Simulation: A survey with research suggestions. AIIE Transaction. 7(3)
20. Tewardt L and Wolkhausan (1989)Th. St. Commun. 70: 839.

APPLICATIONS 1

Intelligent Systems Applied to Optimize Building's Environments Performance

E. Sierra [1], A. Hossian [2], D. Rodríguez [3], M. García-Martínez [4], P. Britos [5], and R. García-Martínez [6]

Abstract By understanding a building as a dynamic entity capable of adapting itself not only to changing environmental conditions but also to occupant's living habits, high standards of comfort and user satisfaction can be achieved. An intelligent system architecture integrating neural networks, expert systems and negotiating agents technologies is designed to optimize intelligent building's performance. Results are promising and encourage further research in the field of AI applications in building automation systems.

1 Introduction

According to the latest definitions internationally accepted for an "intelligent building", this is a building highly adaptable to the changing conditions of its

[1] E. Sierra
Engineering School, Comahue University. esierra@uncoma.edu.ar
[2] A. Hossian
Engineering School, Comahue University. Intelligent Systems Lab. FI-UBA ahossian@uncoma.edu.ar
[3] D. Rodriguez
Software & Knowledge Eng. Center. Buenos Aires Institute of Technology. drodrigu@itba.edu.ar
[4] M. Garcia-Martinez
School of Architecture. University of Buenos Aires. mgm@fadu.uba.ar
[5] P. Britos
Software & Knowledge Eng. Center. Buenos Aires Institute of Technology. pbritos@itba.edu.ar
[6] R. Garcia-Martinez
Software & Knowledge Engineering Center. Buenos Aires Institute of Technology. rgm@itba.edu.ar

Please use the following format when citing this chapter:

Sierra, E., Hossian, A., Rodríguez, D., García-Martínez, M., Britos, P. and García-Martínez, R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 237244.

environment [1]. But, in an overall concept of comfort, the idea of adaptation to changing environmental conditions may be not enough. Building systems are constructed in order to provide comfortable living conditions for the persons who live in them. It is well known that people usually differ in their personal perceptions of comfort conditions. To some extent, the sensation of comfort is an individual one and it is normally affected by cultural issues.

Thus, the idea behind this research is to find techniques based on artificial intelligence in order to provide design recommendations for comfort systems in buildings so that these buildings can also be highly adaptable in terms of the comfort conditions desired by their users. In a few words, a building must "learn" to change its performance not only as a function of environmental conditions, but also as a consequence of preferences set by the people who live in it.

2 The proposed Intelligent System Architecture

According to the latest trends in the field, intelligence in building systems tends to be distributed [2]. The proposed intelligent system architecture is shown in Figure 1. There is a main computer where the functions of monitoring, visualizing and recording parameters is carried out while the regulation functions are left to the local controllers located throughout the building [3]. These controllers are responsible for taking over local control tasks in the zone they serve.

To accomplish its function, the centralized computer contains a database that keeps track of relevant information concerning building user's preferences. For instance, this database keeps records of time, date, number of persons in a room, current temperature and humidity values, as well as temperature and humidity values desired by users. In order to do this, temperature and humidity input panels are located in the different rooms. Each user can eventually set them to what he or she thinks is an ideal comfort condition. As comfort perception is an individual sensation, the database in the main computer keeps track of every individual requirement.

The information contained in the user's requirements database for a given room is applied to a neural network of the self organizational maps of Kohonen (SOM) [4] and [5] type, which is used to cluster all the user's requirements and discard all those groups of requirements which are not relevant in terms of their approximation to the main cluster of preferences. Once a unique group of requirements is selected, their values are applied as input to a program which provides the limits as well as the average value for a particular environmental variable.

This value is used as reference or set-point for the local control strategies set by an expert system which runs on the main computer. This expert system takes decisions concerning control strategies which are used to activate, deactivate or tune the individual controllers. The information about relevant occupancy and setting conditions, as well as the final values of environmental variables is used to train a multi-layer neural network which outcomes will provide ideal

environmental values in case of absence of occupants or of preference information given by them.

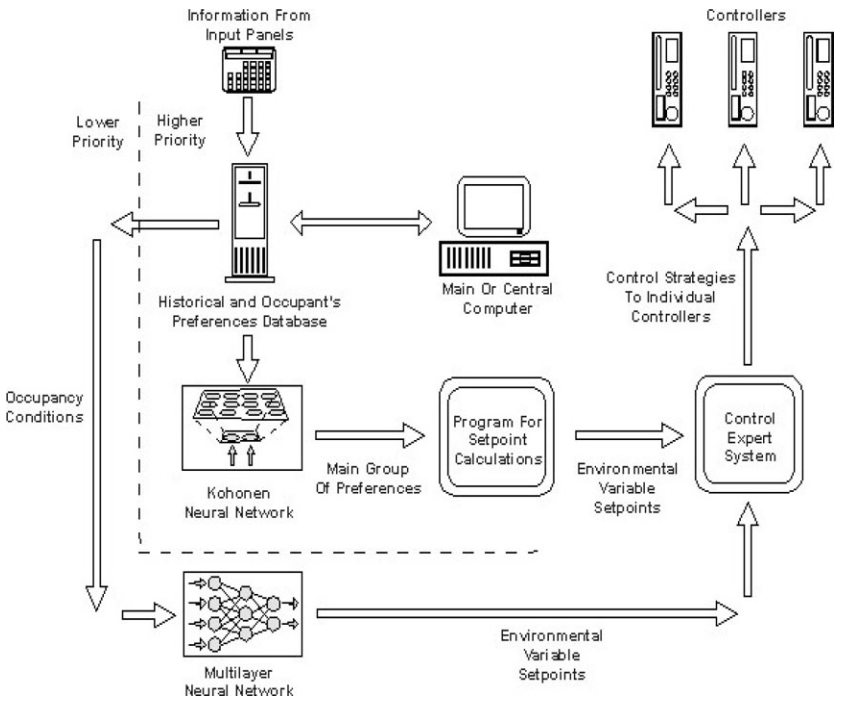


Fig.1. Intelligent System Architecture where the negotiating agent resides in main computer

In any case, set-points assigned to comfort variables provided by the analysis of user's desired environmental conditions is given priority over any automatic calculation of these conditions.

3 Energy saving conditions

A very important issue in intelligent buildings technology is related to energy saving policies [6]. Optimisation procedures carried out to cut off energy consumption rates are not only justified in terms of operation costs reduction but also because of the environmental benefits implied in the adoption of energy saving strategies.

In order to accomplish previously mentioned optimization procedures, an expert system [7] containing rules that perform energy saving strategies is set up in the central computer. However, it is necessary to verify if the rules defined in

the energy saving expert system may eventually alter the comfort conditions established by the control strategy expert system. As it is shown on Figure 2, there is an intelligent negotiation agent [8], [9] and [10] which runs in the central computer created to determine whether the application of energy saving strategies will: a) not affect current comfort conditions in a given space (not affected) b) affect current comfort conditions but within the limits found by the SOM neural network based upon preference information provided by occupants (partially affected) c) affect current comfort conditions beyond the limits set by occupant's requirements (fully affected).

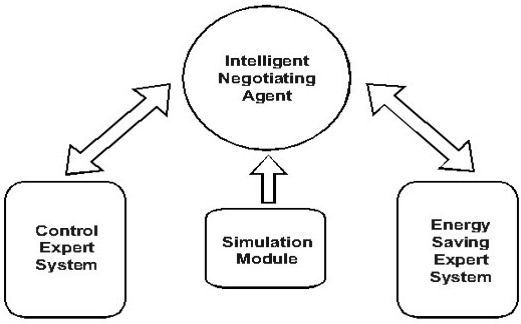


Fig. 2. Negotiation Control and Energy Saving Rules

The policy applied by the intelligent negotiation agent in the different situations mentioned earlier can be summarized as follows:

1. If comfort conditions are not affected, rules defining energy saving strategies are given the highest priority
2. If comfort conditions are partially affected, rules defining energy saving strategies are given an intermediate priority, just lower than the priority given to the rules that regulate the operation of main control actuators.
3. If comfort conditions are fully affected, rules defining energy saving strategies are given the lowest priority.

To be more descriptive in terms of how the inference engine runs, the intelligent negotiation agent was given a number of rules which express the desired energy saving policy (constraints) based on the building conditions. The occurrence of certain events inside the building (e.g. a temperature raises above a permitted upper limit) will trigger the appropriate rule within the agent.

The agent executes the rule(s), with the purpose of readjusting the environmental conditions to some preferred set of values. The triggered rule(s) will cause a set of actions to be immediately executed. After the previously described negotiation policy has been applied, the control expert system located in the main central computer has an updated rule base which can be used to set up the operation mode of local controllers (on, off, normal) and tune them

accordingly, for example, by determining the appropriate set-point for the control variable.

4 An Example

With the purpose of providing an example that illustrates the functionality of the proposed intelligent system, the operation of the air – handling system depicted in Figure 3 will be described. It is assumed that the HVAC engineer has already designed the air handler in terms of laying out the ductwork, appropriately sizing the fan and heating and cooling coils, and selecting the proper dampers, damper actuators and motor contactor. From this design a system diagram has been constructed as shown in Figure 3.

The designations DA and VA stand for damper and valve actuators, respectively, C is for electrical contactor and H/C and C/C represent the heating and cooling coils. When building zone building served by the air –handler is "occupied", i.e., the current date and time fall within a certain schedule, the system is said to be in occupied mode. In this mode, the fan is started and the heating and cooling valves and dampers are modulated so as to maintain the set-point temperature in the zone.

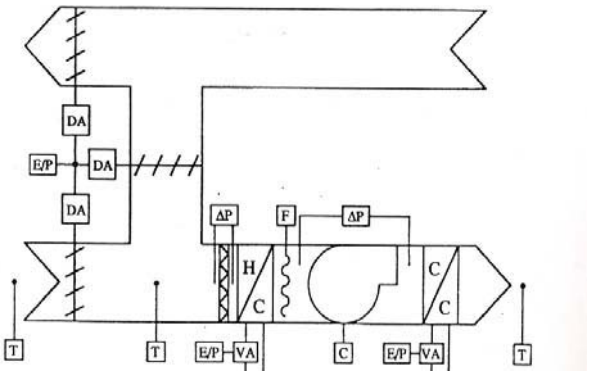


Fig. 3. System Diagram for the Air Handler

This is called the "normal" operating condition. Control strategies describe how specific subsystems are to be controlled. Thus, some of the rules contained in the rule base of the control expert system will stated as follows:

```
IF      the date and time fall within the specified schedule
THEN    the system shall enter the occupied mode.
```

```
IF      the system is in the occupied mode,
THEN    the supply fan shall be turned on,
AND     the normally closed cooling valves and air dampers
        shall be controlled by a sequenced PI (Proportional
        plus Integral) controller to maintain the room air
        temperature set-point to 70 °F.

IF      the date and time fall outside of the specified
        schedule
AND     the room air temperature exceeds 55 °F
THEN    the system shall enter the unoccupied mode.

IF      the system is in the unoccupied mode
THEN    the supply fan shall be turned off, the heating valve
        shall be set to fully open and the cooling valve and
        outside air dampers shall be set to fully closed.

IF      the date and time fall outside of the specified
        schedule
AND     the room air temperature is less than or equal to 55
        °F,
THEN    the system shall enter setback mode.

IF      the system is in the setback mode,
THEN    the system will remain in this mode until the room air
        temperature exceeds 60 °F.
```

Energy saving strategies were designed in order to diminish energy consumption levels while keeping a satisfying response to the building energy demand profiles. Therefore, some of the rules contained in the rule base of the energy saving expert system can be enunciated in the following manner:

Dry Bulb Economizer Control:

```
IF      the system is in occupied mode
AND     the outside air temperature rises above 65 °F, the dry
        bulb economizer set-point
THEN    the outside air damper will be set to a constant
        position of 20%.
```

Mixed Air Low Limit Control:

```
IF      the system is in occupied mode
AND     the mixed air temperature drops from 40 to 30 °F,
THEN    a proportional (P) control algorithm shall modulate
        the outside air dampers from 100 to 0%. Mixed air low
        limit control shall have priority over dry bulb
        economizer control.
```

Free cooling:

```
IF      the system is in unoccupied mode AND the room air
        temperature exceeds 65 °F
AND     the outside air temperature equals to or is less than
        55 °F,
THEN    the supply fan shall be turned on the heating and
        cooling valves shall be set to fully closed and the
        outside air dampers shall be set to fully open.
```

As previously stated, the system tries to capture the occupants' preferences by modifying the set-points of control variables to users' demands.

5 Implementation and Results

A prototype of the proposed intelligent system has been implemented in CLIPS, a tool for developing expert systems. Neural network and negotiating agent algorithms have been programmed in C++. The system prototype has been tested in the building of the Ministry of Education, located in the city of Neuquén, Argentina. This building has been designed with a high degree of intelligence.

After almost a year of continuous tuning and adjusting procedures, the most updated prototype of the system was put to work. The people who work in this public building was strongly encouraged to set comfort parameters in the input control panels that were installed for this purpose in different building zones. The comments of users who admitted positive changes in comfort conditions were confirmed by a survey. The survey outcomes were: 75 % percent of users were very satisfied with the performance of the new system, 20 % were just satisfied and 5% not satisfied. Such results encourage advancing in this direction of optimizing the operative and control strategies carried out by the developed system.

6 Conclusions

Techniques of artificial intelligence have been used in many decision, control and automation systems in the last twenty years. Building systems have not been an exception. In this direction, the intelligent system that is proposed in this article tries to contribute in the field of intelligent buildings optimization, by transforming them in a dynamic space, with high standards of comfort and occupant's satisfaction. In this sense, the ability inherent to intelligent systems that are capable of learning from their own environment plays a very important role in the achievement of these building performance optimization goals. Furthermore, results obtained as a consequence of the proposed system implementation are very encouraging. Thus, further research and development work in the field deserves particular attention.

References

1. Krainier, A. *Toward smart buildings*, Architectural Assn. Graduate School, Environment & Energy Studies Program. (1996).
2. So, A. *Intelligent building systems*, Kluwer Academic Press. (1999).

3. Wong, K. *The Intelligent Building Index: IBI manual: version 2.0*, Hong Kong: Asian Institute of Intelligent Buildings. (2001).
4. Rich E. and Kevin, K. *Introduction to Artificial Networks*. Mac Graw-Hill. Publications. (1991).
5. Hilera J. and Martínez V. *Redes Neuronales Artificiales. Fundamentos, modelos y aplicaciones*. RA-MA, Madrid. (1995).
6. Sierra, E., Hossian, A., Labriola, C., García Martínez R. *Optimal Design of Constructions: A Preliminary Model.* Proceedings of the World Renewable Energy Congress (WREC 2004) – Denver, Colorado, Estados Unidos. (2004).
7. García Martínez, R. and Britos, P. *Ingeniería de Sistemas Expertos*. Editorial Nueva Librería. 649 páginas. ISBN 987-1104-15-4. (2004).
8. Allen, J. F, Kautz, H., Pelavin, R. N., and Tenenberg, J.D. *Reasoning About Plans*. Morgan Kaufmann Publishers, Inc. San Mateo, California. (1991).
9. Conry S. E., Meyer R. A., and Lesser V.R. *Multistage negotiation in distributed planning*. En Bond A and Gasser, L. [Eds] Readings in Distributed Artificial Intelligence. Morgan Kaufmann Publishers, Inc. San Mateo, California. (1988).
10. Ferber J. and Drougol A. *Using reactive multiagent systems in simulation and problem solving*. In Avouris, N.M and Gasser L. [Eds], Distributed Artificial Intelligence: Theory and Praxis. Kluwer Academic Press. (1992).

A Comparative Analysis of One-class Structural Risk Minimization by Support Vector Machines and Nearest Neighbor Rule

George G. Cabral and Adriano L. I. Oliveira

Department of Computing and Systems,Polytechnic School of Pernambuco, University of Pernambuco, Rua Benfica, 455, Madalena, 50.750-410, Recife-PE, Brazil{ggc,adriano}@dsc.upe.br

One-class classification is an important problem with applications in several different areas such as outlier detection and machine monitoring. In this paper we propose a novel method for one-class classification, referred to as kernel k-NNDDSRM. This is a modification of an earlier algorithm, the kNNDDSRM, which aims to make the method able to build more flexible descriptions with the use of the kernel trick. This modification does not affect the algorithm's main feature which is the significant reduction in the number of stored prototypes in comparison to NNDD. Aiming to assess the results, we carried out experiments with synthetic and real data to compare the method with the support vector data description (SVDD) method. The experimental results show that our one-class classification approach outperformed SVDD in terms of the area under the receiver operating characteristic (ROC) curve in six out of eight data sets. The results also show that the kernel kNNDDSRM remarkably outperformed kNNDDSRM.

1 Introduction

One-class classification differs from normal classification because in the training phase there are data samples from only one class available to build the model [5][9][10][11]. The term one-class classification originates from Moya [12], but also outlier detection [13], novelty detection [2] or concept learning [7] are used.

Outlier detection is the task of learning what is normal and determining when an event occurs that differs significantly from expected normal behavior. The approach that outlier detection takes is the opposite of signature detection (which can be implemented using multi-class classification). Signature detection is explicitly given information on what is novelty, and simply attempts to detect it when it happens. False alarms are rare when using signature detection because the algorithm has been programmed to know exactly what to look for to detect the known novelty conditions. However, signature detection is unable to detect *new* unknown events. Although outlier detection systems produce more false

Please use the following format when citing this chapter:

Cabral, G.G. and Oliveira, A.L.I., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 245254.

alarms than signature detection systems, they have the significant advantage that they are able to detect new, previously unknown, novelty behavior [14].

Structural risk minimization (SRM) [16] aims to find the function that for a fixed amount of data achieves the minimum of guaranteed risk. In our approach we do not search for a function that best fit the data, we try to find the more representative and smaller amount of data in the training set according with the empirical risk minimization principle (ERM). Many other approaches for multi-class classification have a similar goal. An example is a method to prune neurons from a neural network which have similar outputs given the same input aiming to reduce the complexity of the network.

In a recent paper, we proposed to implement one-class classification with the SRM principle using a nearest neighbor (NN) rule, referred to as k-NNDDSRM [4]. One of the objectives of k-NNDDSRM is to reduce the number of instances in an NNDD like one-class classifier while improving its classification performance. Analysis has shown that this new method had a lower complexity in comparison with the NNDD [15] with an improved performance in almost all data sets considered in the experiments [3, 4].

In this paper we propose a modification in the original k-NNDDSRM to make the one-class classifier able to work in a non-Euclidean space through the use of kernel operators. The novel method introduced in this paper is referred to as *kernel k-NNDDSRM*. The idea is to map the original input space into an n-dimensional hyperspace. By doing this we establish a connection between SVM classification and our NN rule. We also make a structural changing in the original algorithm by eliminating the concept of *center of mass*, proposed in [3], thereby introducing a more general form to build the data description.

To evaluate the effectiveness of our proposed method we conducted some experiments using both artificial and real-world data sets and compared it with both the SVDD [15], Support Vector Data Description, and the original k-NNDDSRM [4]. In this paper we have chosen the SVDD by its SVM nature which means we are dealing with one of the more sophisticated and powerful methods available today. Performance is assessed by calculating the receiver operating characteristics (ROC) curves and computing the AUCs (Areas Under the Curves).

Next section briefly reviews the Support Vector Data Description method for one-class classification. Section 3 details the proposed modification in the k-NNDDSRM, named kernel k-NNDDSRM. Section 4 presents the experiments and the results, including a comparison with SVDD and the original k-NNDDSRM. Finally in section 5 conclusions and suggestions for further research are presented.

2 Support Vector Data Description - SVDD

Support vector machines (SVMs) comprise state of the art machine learning methods based on the principle of structural risk minimization (SRM) [16]. SVMs can be applied, for instance, for classification and regression. SVM is one of the most sophisticated nonparametric supervised classifiers available.

One-class SVM works by mapping the data onto the surface of a hyper sphere in the feature space. The goal is to maximize the margin of separation from the origin. This is equivalent to Support Vector Data Description (SVDD)[15] which finds the smallest sphere enclosing the data. As in multi-class SVMs, slack variables, denoted by $\boldsymbol{\xi}_i$, are associated to each data sample. This allows the possibility that some of the training data samples fall outside the description (i.e. are misclassified as outliers) when the minimum radius is found.

Fig. 1 shows an example in which a data description is built and 3 objects reside in the boundary of the description and one, with $\boldsymbol{\xi}_i > 0$, falls outside of the description. These 4 objects are called support vectors.

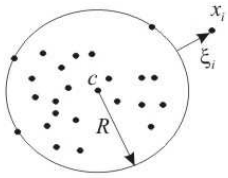


Fig. 1 Hypersphere Generated by SVDD

Let $\phi : X \rightarrow H$ be a kernel map which transforms the training sample from a space X to another space H. To separate the data from the origin with maximum margin one needs to solve the following quadratic problem:

$$\mathbf{min} \frac{1}{2}\|\omega\|^2 - \rho + \frac{1}{\nu\ell}\sum_{i=1}^{l}\boldsymbol{\xi}_i \tag{1}$$

where ω is the normal vector to the separating hyper plane, ℓ is the number of training samples and ρ is the offset, subject to $(\omega \bullet \Phi(x_i)) \geq \rho - \xi_i \quad i = 1, 2, ..., \ell \quad \xi_i \geq 0$.

If ω and ρ solve this problem, then we have found a function $f(x) = sign\left((\omega \bullet \Phi(x))\right) - \rho)$ such that if $f(x) > 0$, the object x is classified as normal. Otherwise, x is classified as novelty.

When $\rho > 0$ then the parameter $\upsilon \in (0, 1)$ is an upper bound on the fraction of outliers (i.e. training error) and also a lower bound on the fraction of support vectors. The dual problem is: $\mathbf{min}_\alpha \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j k(\Phi(x_i), \Phi(x_j))$, subject to $0 \leq \alpha_i \leq \frac{1}{\nu\ell} \quad and \quad \sum_i \alpha_i = 1$. Now the decision function is

$$f(x) = sign\left(\sum_{i=1}^{\ell} \alpha_i k(\Phi(x_i), \Phi(z)) - \rho\right) \tag{2}$$

and ρ can be recovered by

$$\rho = \sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j k(\Phi(x_i), \Phi(x_j)) \tag{3}$$

where $0 \leq \alpha_i, \alpha_j \leq \frac{1}{\nu\ell}$.

To carry out simulations using SVDD in our research, we have used the DD_Tools (Data Description toolbox) version 1.6.1. This is an integrated tool for one-class classification which can handle a number of one-class classification algorithms. The DD_Tools 1.6.1 is available at http://www-ict.ewi.tudelft.nl/~davidt/dd_tools.html. In DD_Tools the parameter ν is replaced by the *fracrej* parameter, which gives the fraction of the training set which will be rejected. Therefore, in the experiments we will only refer to the parameter *fracrej*. The parameters used in our experiments will be explained in Section 4.

3 Kernel *k*-Nearest Neighbor Data Description with Structural Risk Minimization - kernel k-NNDDSRM

In this Section we first explain how the training phase of the kernel NNDDSRM is performed and then we show how the kernel k-NNDDSRM uses the kernel NNDDSRM to classify objects taking into account the k nearest neighbors.

3.1 Kernel NNDDSRM

The main feature of the NNDDSRM [3] consists of reducing the number of stored prototypes. This reduction produces at least two improvements. The first improvement is a reduction in the search time for neighbors in the classification phase. The second improvement is the reduction in memory space for data storage.

NNDDSRM is based on NNSRM [8], a classification algorithm based on NN (Nearest Neighbor) and SRM. The idea of NNSRM for the case of one-class classification is to include in the prototype set only the training samples which are in the harder region for classification. The training samples are included in the prototype set until the training error becomes zero.

The first step of the *kernel NNDDSRM* consists of computing a matrix $n \times n$, where n is the number of input patterns in the training set, with the results of

the kernel function for each two input patterns. After computing the matrix we compute an array containing a sum, *Si*, of each row as shown in Eq. 4.

$$\begin{array}{l} \sum_{i=1}^{\ell} k(x_i, x_1) = s_1 \\ \sum_{i=1}^{\ell} k(x_i, x_2) = s_2 \\ \quad ... \qquad\quad ... \; ... \\ \sum_{i=1}^{\ell} k(x_i, x_l) = s_l \end{array} \tag{4}$$

For this work we have used the RBF kernel (Eq. 5).

$$K(x_i, x_j) = \exp\left(\frac{- \parallel x_i - x_j \parallel^2}{\sigma^2}\right) \tag{5}$$

In Eq. 5, the σ value is not a crucial parameter to obtain a good kernel kNNDDSRM classifier. We have performed several experiments and varied σ; the results have shown that σ has no significant influence on performance.

After computing the array S, containing the s_i's (Eq. 4), it must be sorted in ascending order.

In the training phase, the kernel NNDDSRM will compute two different sets of samples, namely, the rejected set (RS) and the prototype set (PS). RS contains the *fracrej* patterns with smallest s_i. The idea is that a fraction of the training set (*fracrej*) should be considered outliers. On the other hand, PS is a set which stores prototypes that delimit the region of normal patterns. The inner training samples, that is, those with greatest sum s_i, will not be included in PS. The number of samples to be stored in PS is determined as in NNSRM, that is, training samples are included in PS as needed to make the training error equal to zero.

After training, we have two sets of training samples, namely, PS (Prototype Set) and RS (Rejected Set). Both sets are used in the test phase of the algorithm, therefore the total number of prototypes stored by the algorithm is the number of samples in PS plus the number of samples in RS.

The following pseudo-code shows the training phase of NNDDSRM.

1. Load data of the training set (TS)
2. Compute the array (S) containing all the summing of each RBF between each input sample and the rest of the samples
3. Sort TS in increasing order, according with S.
4. Remove *fracrej*% of the samples from the beginning of TS and add them to RS
5. Remove the two first samples in TS and add to PS.
6. FOR ALL training pattern (p)
   d1 = max(K(p,q) | $q \in RS$)
   d2 = max(K(p,j) | $j \in PS$)
   IF (d2/d1) < 1
   errorCounter++
7. IF errorCounter > 0

```
        //Remove the 2 first patterns from TS, add into PS, reset errorCounter
        //and go back to (7)
    ELSE
        //End
```

The test phase, for a prototype p, is performed using the following pseudo-code:

```
r1 = max(K(p,RS))
r2 = max(K(p,PS))
if (r2/r1) < th
    return NOVELTY
else
    return NORMAL
```

3.2 Kernel k*-NNDDSRM*

The kernel k-NNDDSRM method consists solely of an extension of the kernel NNDDSRM involving the k members from PS and RS with highest kernel outputs for given test object. The kernel output of the first prototype with highest kernel output in PS is compared to the kernel output of the first prototype in RS with highest kernel output to a test object. The comparison is repeated for the next $k-1$ prototypes in PS and RS with highest kernel outputs for a given test object.

The following pseudo-code shows how the algorithm takes a decision on a pattern z to be classified:

```
1. kRS // set with the k prototypes with highest kernel outputs to z in
       //RS, increasing order
2. kPS // set with the k prototypes with highest kernel outputs to z in
       // PS, increasing order
   NOVELTIES = 0 // number of patterns classified as novelties
   NORMAL = 0 // number of patterns classified as normal
3. for ( i = 1 to i ≤ k)
       d1 = K(z,kRS[i])
       d2 = K(z,kPS[i])
           IF (d1/d2) ≤ th
               normal++;
           ELSE
               novelties++;
   end for
4. IF(novelties ≤ normal)
       //the pattern z is classified as novelty
   ELSE
       //the pattern z is classified as normal
```

4 Experiments

This section reports on experiments carried out to evaluate the performance of the kernel k-NNDDSRM method and to compare it to SVDD and k-NNDDSRM. For the experiments with the three methods we considered a range of 5% to 25% for the *fracrej* parameter. The parameter k of the kernel k-NNDDSRM and k-NNDDSRM were varied from 1 to 5 and the parameter σ of the SVDD method was varied with values [5, 10, 15, 20].

To evaluate the methods we have used the area under the curve (AUC) produced by the receiver operating characteristic curves (ROC) which is frequently used to evaluate one-class classifiers and methods for novelty detection [15], [5], [14]. In the ROC curve, the x-axis represents the PFA (Probability of False Alarm), which identifies normal patterns wrongly classified as novelties; the y-axis represents the PD (Probability of Detection), which identifies the probability that patterns of the novelty class be recognized correctly. The ROC curve depicts several operating points where each one of these operating point consist of a different classifier.

Aiming to obtain the most accurate points to build the ROC curve we have generated an array having length = #(test dataset), containing all the results values of testing the model in each sample from the test dataset. After creating the array, we sorted this array in increasing order and applied the same approach used for Tax [15] for building the ROC curve; this approach achieves the most accurate points with a low computational cost. With this approach we do not need to vary any parameter for building the ROC curves.

The experiments were conducted using six data sets, three of them from the UCI repository [1]. We have used two artificial data sets and four real world data sets in the experiments.

The first artificial data set was generated from two Gaussian Distributions and was also used in [3][4]. In the Gaussian Distributions data set the samples belonging to normal class were generated by a Gaussian distribution with mean 0 and covariance 4 and the samples belonging to novel class by one with mean 4 and covariance 4. This data set is particularly important because it is visually possible to analyze the behavior of the algorithm and to validate it.

The Banana Shaped data set, as the previous data set, is an artificial bi-dimensional data set which was also used in [15]. This data set was generated with the prtools Matlab toolbox [6].

Fig. 2 shows a small, but representative, fraction of the samples of the bi-dimensional Gaussian Distributions data set and of the Banana Shaped data set.

Three of the real-world data sets were obtained from the UCI Repository [1]: (1) Iris, (2) Winscounsin Breast Cancer and (3) Pima Indian Diabetes. The breast cancer and diabetes are two classes data sets. The Iris data set has three different classes, thus we generated three different data sets from it for novelty detection experiments. In each data set, a different class was selected to represent novelties whereas patterns from the remaining classes represented

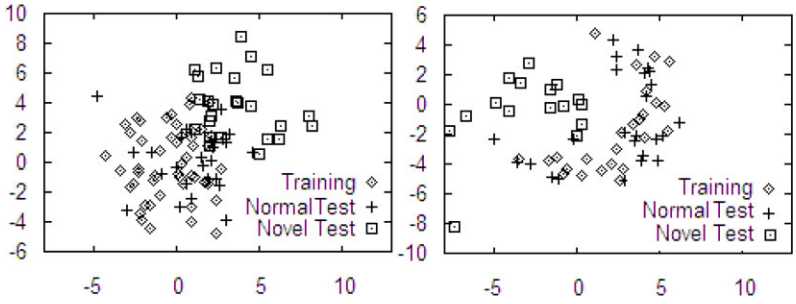


Fig. 2 Synthetic Gaussian Distributions and Banana data sets distribution

the normal class. For simplicity we labeled class Iris-setosa as 1, Iris-versicolor as 2 and Iris-virginica as 3. Thus we generated three different data sets, named *Iris class 1*, *Iris class 2*, and *Iris class 3*. The Biomed data set, available in the StatLib (http://lib.stat.cmu.edu/datasets) archive, was also used in our experiments. This data set was also used in [5]. Table 1 shows the partitioning of the data sets used in the experiments.

Table 1 Data sets Patterns Partitioning

Data set	♯Training Patterns	Test Patterns	
		♯normal	♯novelty
Gaussian Distributions	300	150	150
Banana	80	80	80
Iris class 1	50	50	50
Iris class 2	50	50	50
Iris class 3	50	50	50
Diabetes	250	250	268
Breast Cancer	184	269	239
Biomed	80	54	75

Table 2 shows the results of the comparison of both methods, kernel k-NNDDSRM and SVDD. The best AUC results are shown in boldface.

Table 2 kernel k-NNDDSRM and SVDD results

Data set	Kernel k-NNDDSRM					SVDD				
	fracrej%	k	#prot	%Total	AUC	fracrej%	σ	#SV	%Total	AUC
Gaussian Distributions	17	2	85	28.33	0.9144	8	10	226	75.33	**0.9351**
Banana	24	4	28	35	0.9309	10	5	11	13.75	**0.9864**
Iris Class 1	20	2	17	34	**1.0**	5	5	4	8	0.9800
Iris Class 2	6	1	5	10	**0.5910**	8	5	6	12	0.1296
Iris Class 3	16	3	19	38	**0.9848**	6	5	6	12	0.9736
Biomed	13	4	21	26.25	**0.9080**	5	5	80	100	0.8725
Diabetes	19	2	157	62.8	**0.7017**	7	20	167	66.8	0.6548
Breast Cancer	20	2	86	46.73	**0.9974**	7	5	61	33.15	0.7781

For both synthetic data sets the SVDD slightly outperformed our proposed method. For the synthetic Gaussian Distributions data set the best result when using kernel k-NNDDSRM was achieved using the parameter fracrej set to 17% and $k = 2$. In this case, we observed a performance loss, relative to SVDD, of 2.07%, on other hand only 28.33% of the entire training set was used for classification whereas the best SVDD used 75.33%.

Our proposed method outperformed the SVDD in all four real world data sets of Table 2. In the Iris data set, when the class 1 was elected as novelty, we have achieved the best possible result, AUC = 1. With class 2 as novelty we achieved a poor result with both methods. In the Diabetes data set, even achieving a considerably better result than the SVDD, the AUC of 0.7017 was not satisfactory. In the Biomed data set the kernel k-NNDDSRM has achieved a better AUC than the SVDD storing 73.75% less prototypes. A great performance was also achieved in the Breast Cancer data set. An AUC of 0.9974 was achieved by our proposed method storing only 46.73% of the entire training set.

Finally, we compare the performance of the kernel k-NNDDSRM with our earlier method, the k-NNDDSRM [4]. Table 3 shows the best results obtained in this paper and in [4], considering the same data sets. Once more, the boldface AUCs show the bests results. The results show that the kernel k-NNDDSRM remarkably outperformed the original k-NNDDSRM in the first three data set and obtained similar result in the last one.

Table 3 kernel k-NNDDSRM and standard k-NNDDSRM results [4]

Data set	Kernel k-NNDDSRM					k-NNDDSRM				
	fracrej%	k	#prot	%Total	AUC	fracrej%	k	#prot	%Total	AUC
Gaussian Distributions	17	2	85	28.33	**0.9144**	5	1	40	13.3	0.7640
Biomed	13	4	21	26.25	**0.9080**	15	3	21	26.25	0.8500
Diabetes	19	2	157	62.8	**0.7017**	25	9	157	62.8	0.6470
Breast Cancer	20	2	86	46.73	**0.9974**	15	3	125	50	0.9950

5 Conclusion

In this paper we proposed a novel method for one-class classification named kernel k-NNDDSRM. It is a modification of an early method that we developed, the k-NNDDSRM. The new method aims to obtain more flexible descriptions than a sphere shaped description, achieved by the original k-NNDDSRM. This was done by using the kernel trick in our method and also by eliminating the concept of *center of mass* [3, 4]. Both methods have a parameter k which makes the final result more dependent on the neighborhood [4].

The novel method was able to achieve a significant reduction in the number of stored prototypes in comparison to NNDD, which stores all training patterns.

This reduction is directly related to the parameter *fracrej* which indicates the fraction of prototypes in training set that should fall outside the description boundary.

Our simulations using real and synthetic data sets have shown that the proposed method has achieved a good performance in comparison with the SVDD method. In six out of eight data sets our method outperformed SVDD. In comparison with the original k-NNDDSRM our method has obtained much better results in all data sets.

Our future work will include the use of other kernels besides the RBF kernel. We also aim to adapt our method for training with examples of the novelty class as well as of the normal class, as in [15].

References

1. Asuncion, A., Newman, D.: UCI machine learning repository (2007). URL `www.ics.uci.edu/~mlearn/MLRepository.html`
2. Bishop, C.M.: Novelty detection and neural network validation. IEE Proceedings — Vision, Image and Signal Processing **141**(4), 217–222 (1994). Document No. 19941330
3. Cabral, G.G., Oliveira, A.L.I., Cahú, C.B.G.: A novel method for one-class classification based on the nearest neighbor rule and structural risk minimization. In: Proc. IJCNN'2007, International Joint Conference on Neural Networks, pp. 1976–1981 (2007)
4. Cabral, G.G., Oliveira, A.L.I., Cahú, C.B.G.: Combining nearest neighbor data description and structural risk minimization for one-class classification. Neural Computing & Applications (2008). Accepted for publication
5. Cao, L., Lee, H.P., Chong, W.K.: Modified support vector novelty detector using training data with outliers. Pattern Recognition Letters **24**(14), 2479–2487 (2003)
6. Duin, R.P.W., Juszczak, P., de Ridder, D., Paclík, P., Pekalska, E., Tax, D.M.J.: PR-Tools 4.0, a Matlab toolbox for pattern recognition. http://www.prtools.org (2004)
7. Hanson, S.J., new Brunswick, G.S., Kulikowski, C., Japkowicz, N.: Concept-learning in the absence of counter-examples: An autoassociation-based approach to classification. Tech. rep. (1999). URL `http://citeseer.ist.psu.edu/222433.html`
8. Karacali, B., Krim, H.: Fast minimization of structural risk by nearest neighbor rule. IEEE Trans. on Neural Networks **14**, 127–137 (2003)
9. Markou, M., Singh, S.: Novelty detection: a review - part 1: statistical approaches. Signal Processing **83**(12), 2481–2497 (2003)
10. Markou, M., Singh, S.: Novelty detection: a review-part 2: neural network based approaches. Signal Processing. Dec. 2003; 83(12): 2499-521 (2003)
11. Marsland, S., Nehmzow, U., Shapiro, J.: On-line novelty detection for autonomous mobile robots. Robotics and Autonomous Systems **51**(2-3), 191–206 (2005)
12. Moya, M.M., Koch, M.W., Hostetler, L.D.: One-class classifier networks for target recognition applications. In: Proc. WCNN'93, World Congress on Neural Networks, vol. III, pp. 797–801. INNS, Lawrence Erlbaum, Hillsdale, NJ (1993)
13. Ritter, G., Gallegos, M.T.: Outliers in statistical pattern recognition and an application to automatic chromosome classification. Pattern Recognition Letters **18**(6), 525–539 (1997)
14. Salvador, S.W.: Learning states for detecting anomalies in time series. Master's thesis, Florida Institute of Technology (2004)
15. Tax, D.M.J.: One-class classificationconcept-learning in the absence of counter-examples. Ph.D. thesis, Technische Universiteit Delft (2001)
16. Vapnik, V.: Statistical Learning Theory. Wiley (1998)

Estimation of the Particle Size Distribution of a Latex using a General Regression Neural Network

G. Stegmayer, J. Vega, L. Gugliotta, and O. Chiotti

Abstract This paper presents a neural-based model for estimating the particle size distribution (PSD) of a polymer latex, which is an important physical characteristic that determines some end-use properties of the material (e.g., when it is used as an adhesive, a coating, or an ink). The PSD of a dilute latex is estimated from combined DLS (dynamic light scattering) and ELS (elastic light scattering) measurements, taken at several angles. To this effect, a neural network approach is used as a tool for solving the involved inverse problem. The method utilizes a general regression neural network (GRNN), which is able to estimate the PSD on the basis of both the average intensity of the scattered light in the ELS experiments, and the average diameters calculated from the DLS measurements. The GRNN was trained with a large set of measurements simulated from typical asymmetric PSDs, represented by unimodal normal-logarithmic distributions of variable geometric mean diameters and variances. The proposed approach was successfully evaluated on the basis of both simulated and experimental examples.

G. Stegmayer
CIDISI-CONICET, Lavaise 610 - 3000 - Santa Fe, Argentina e-mail: gstegmayer@santafe-conicet.gov.ar

J. Vega
CIDISI-CONICET and INTEC-CONICET, Güemes 3450 - 3000 - Santa Fe, Argentina e-mail: jvega@santafe-conicet.gov.ar

L. Gugliotta
INTEC-CONICET, Güemes 3450 - 3000 - Santa Fe, Argentina e-mail: lgugliott@santafe-conicet.gov.ar

O. Chiotti
CIDISI-CONICET and INGAR-CONICET, Avellaneda 3657 - 3000 - Santa Fe, Argentina e-mail: chiotti@santafe-conicet.gov.ar

Please use the following format when citing this chapter:

Stegmayer, G., Vega, J., Gugliotta, L. and Chiotti, O., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 255264.

1 Introduction

Polymers play a major role in the current production of materials, both mass consumer commodities (such as engineering plastics, rubber, etc.) and more special products (adhesives, paints and coatings, reagents for medical diagnosis, etc.) [1]. The production of polymers with pre-specified quality characteristics is an important scientific and technological challenge, which combines expertise in at least two major research areas: a) optimization of production processes, and b) characterization of the obtained products. The first line intends to define the best way to produce the polymer, and involves the development of estimation, optimization, and control techniques, usually based on mathematical models representing the process [2]. The second line is intended to determine the quality of a product, using analytical specific techniques and physical, chemical or mechanical tests on the properties of the final product [3]. Nowadays, it is possible to simulate detailed mathematical models of the process dynamics, which can easily involve dozens of simultaneous differential and algebraic equations [4]. At an early stage, the model parameters are adjusted off-line to the main process variable measurements. Subsequently, the adjusted model can be used to design operation and control strategies that may enable an optimum polymer production with pre-specified quality characteristics.

Product characterization involves standard procedures for signals analysis and data treatments. In this case, it is usually necessary to solve ill-conditioned inverse problems, which result from indirect measurements of the desired properties, combined with theoretical principles of the employed analytical techniques [5]. The resolution of such problems involves the use of numerical techniques for digital filtering and functions regularization, to partially mitigate the inevitable noise measurement presence in the signals and systematic mistakes committed during the modelling of the associated analytical technique, which limits the accuracy and resolution of the obtained solutions. As an alternative to detailed models, artificial neural networks (NN) allow describing the system from the viewpoint of their input/output behavior [6]. A NN appropriately adjusted to a given process allows variables estimation in short times, thereby facilitating its subsequent implementation in on-line control process strategies [7].

Regarding analytical techniques for polymeric end-products characterization, the resulting problems are hard to solve due to: (i) the indirect and relative characteristics of the involved measurements, (ii) the low information content in the measurements regarding the properties of interest, and (iii) the need of solving an ill-conditioned inverse problem. For example, the quality of some polymer colloids (or latexes) are normally associated to their particle size distributions (PSD). Such characteristic determines some end-use properties (e.g., rheological, mechanical, and physical properties) of the material when used as an adhesive, a coating, or an ink. For example, the PSD can define the behavior of adhesives and paints, and the chemical stability of latexes; and it can influence the physico-chemical mechanisms involved in emulsion polymerization [8]. Unfortunately, there is no analytical instrumentation capable of directly measuring a PSD. For this reason indirect measurements are needed, where the measured physical variables are related to the PSD

through theoretical models. Some optical techniques, such as elastic light scattering (ELS) or dynamic light scattering (DLS), can estimate a latex PSD from measurements of the light scattered by particles in dispersion, when they are lightened with a monochromatic light (typically, a laser). These techniques are sustained in the Mie theory, which describes the light scattered by a particle at different measured angles [9]. The resolution of the resulting inverse problem is usually approached using standard regularization techniques [10], but the obtained solutions have low resolution (i.e., inability to differentiate among similar particles). The combination of measurements tends to increase the information content of the property to be measured. To improve the estimation of a latex PSD, some progress has been made by combining ELS and DLS measurements carried out at multiple angles [11], even if the refractive index of the particles is unknown [12].

The application of NN for the resolution of an inverse problem associated to characterization techniques is scarce. For example, NNs have been used for pattern recognition in high performance liquid chromatography [13]. They have also been used to estimate: a) the radius and refractive index of homogeneous spherical particles, based on a reduced number of light scattering measurements taken at multiple angles [14], b) the PSD of an aerosol, from measurements of laser light diffraction [15], and c) the radius, aspect ratio, and orientation of cylindrical and spherical particles, from light scattering measurements at multiple angles [16]. This paper proposes the use of a NN for the resolution of an ill-conditioned inverse problem as an effective tool to mitigate the effect of noise on measurements; and to achieve better solutions than those obtained through classical inversion procedures. To date, it is unknown the existence of research papers using a NN for estimating a latex PSD from combined DLS and ELS measurements.

The organization of this work is the following: Section 2 introduces some fundamentals concepts of DLS and ELS measurement techniques; Section 3 explains the proposed neural network-based inverse model; Section 4 presents some simulation and experimental results for model validation, and finally, Section 5 summarizes the main conclusions of the work.

2 DLS and ELS fundamentals

Both DLS and ELS are optical techniques widely used for measuring mean diameters and PSD of polymer latexes in the sub-micrometer range. The instruments employed for DLS and ELS techniques basically consist of: i) a monochromatic laser light that falls onto a dilute latex sample; and ii) a photometer placed at a given detection angle, θ_r, with respect to the incident light, that collects the light scattered by the particles over a small solid angle. In practice, DLS and ELS have been broadly employed for measuring mean diameters and PSD of polymer latexes [17]. The PSD is calculated by solving an ill-conditioned inverse problem, on the basis of a mathematical model describing the light scattering phenomena (e.g, the Mie theory [18] [19]). Unfortunately, single optical measurements have a low information content

on the PSD; and consequently only a rather poor PSD resolution is expected. The combination of two or more independent sets of measurements allows increasing the information content, and can contribute to improve the quality of the PSD estimate [20][21].

A photometer placed at θ_r collects the light scattered by particles in a diluted latex sample. In ELS, the light intensity $I(\theta_r)$ is measured at each angle θ_r. In DLS, a dedicated digital correlator, together with special software, measures the first order autocorrelation function of the light scattered at every θ_r, $g^{(1)}_{\theta_r}(\tau)$, for different values of the time delay τ [9]. For each θ_r $(r = 1,2,\cdots R)$, the measurements model can be described through the following first order Fredholm equations [11][12]:

$$I(\theta_r) = \int_0^\infty C_I(\theta_r, D) f(D) dD; \quad r = 1, \cdots, R \tag{1}$$

$$g^{(1)}_{\theta_r}(\tau) = \int_0^\infty e^{-\frac{\Gamma_0(\theta_r)}{D}} C_I(\theta_r, D) f(D) dD; \;\; r = 1, \cdots, R \tag{2}$$

where $f(D)$ is the unknown PSD, represented by the number of particles with diameter D; $C_I(\theta_r, D)$ is the light intensity scattered by a particle with diameter D at θ_r calculated through the Mie theory, and $\Gamma_0(\theta_r)$ depends on several experimental conditions [11]. In general, the estimation problem consists in finding the (unknown) $f(D)$ by inverting equations 1 and 2. Such inverse problem is normally ill-conditioned; i.e., small errors in the measurement (for example, small perturbations due to measurement noise) can originate large changes in the $f(D)$ estimate. Moreover, the difficulty of the inverse problem increases as the distribution becomes narrower.

While DLS is reliable and fast for evaluating average particle diameters, it exhibits serious limitations for estimating the PSD due to the extreme ill-conditioning of equation 2, that makes it impossible to exactly obtain the PSD by numerical methods. Regularization methods aim at improving the numerical inversion by including adjustable parameters, a priori knowledge of the solution, or some smoothness conditions [10]. While a strong regularization produces an excessively smoothened and wide PSD, a weak regularization normally originates oscillatory PSD estimates. Thus, a trade-off solution must be selected. In general, the estimation of a narrow PSD is more difficult than the estimation of a wide PSD.

The combination of independent measurements allows increasing the information content and can contribute to improve the quality of the estimated PSD [12]. Properly combining the previous equations, an inverse problem can be stated for estimating the PSD of a latex from ELS and DLS measurements. This approach proposes to combine, for each θ_r, a scalar value $I(\theta_r)$ with a function $g^{(1)}_{\theta_r}(\tau)$. However, both independent problems (ELS and DLS) are known to be ill-conditioned; therefore their combination into one problem will also be ill-conditioned. To overcome this problem, we propose to replace the equation 2 by the mean diameter calculated with DLS measurements at each θ_r. That diameter - which we will call $D_{DLS}(\theta_r)$ - can accurately be evaluated in most commercial equipment. For a given PSD, $D_{DLS}(\theta_r)$ is calculated through:

$$D_{DLS}(\theta_r) = \frac{\int_0^\infty C_I(\theta_r, D) f(D) dD}{\int_0^\infty \frac{C_I(\theta_r, D) f(D)}{D} dD}; \quad r = 1, \cdots, R \tag{3}$$

Call $f(D_i)$ the discrete number PSD, where f represents the number of particles contained in the diameter interval $[D_i, D_{i+1}]$, with $i = 1, 2 \cdots, N$. All the D_i values are spaced at regular intervals ΔD along the diameter range $[D_{min}, D_{max}]$; thus, $D_i = D_{min} + (i-1)\Delta D$, with $\Delta D = (D_{max} - D_{min})/(N-1)$.

Now equations 1 and 3 may be re-written as:

$$I(\theta_r) = \sum_{i=1}^{N} C_I(\theta_r, D_i) f(D_i); \;\; r = 1, \cdots, R \tag{4}$$

$$D_{DLS}(\theta_r) = \frac{\sum_{i=1}^{N} C_I(\theta_r, D_i) f(D_i)}{\sum_{i=1}^{N} \frac{C_I(\theta_r, D_i) f(D_i)}{D_i}}; \;\; r = 1, \cdots, R \tag{5}$$

Then, the estimation problem consists in finding the PSD ordinates $f(D_i)$, by inverting equations 4 and 5.

3 The proposed inverse neural model

To estimate the PSD from the indirect measurements $I(\theta_r)$ and $D_{DLS}(\theta_r)$ the ill-conditioned non-linear inverse problem of equations 4 and 5 must be solved. To avoid solving such difficult problem, this work proposes the estimation of $f(D_i)$ through a NN-based model. To this effect, a general regression neural network (GRNN) is employed [22]. A GRNN is a normalized radial basis function network in which there is a hidden unit (k) centered at every learning case [23]. In a GRNN the number of neurons equals the total number of input/target vectors (K) selected for building the model. The hidden-to-output weights (w_{ik}) are just the target values, so the output is simply a weighted average of those target values that are close to the given input case. Strictly, a GRNN model is directly built on the basis of the learning cases, and therefore no specific training algorithm is required. The GRNN can also be considered as a one-pass learning algorithm, with a highly parallel structure. Even with sparse data in a multidimensional measurement space, the algorithm provides smooth transitions from one observed value to another [22].

When using GRNN models, the selection of an appropriate smoothing (spread) factor is required, to be applied to each of the radial units, which indicates how quickly the activation function decreases as the distance increases from the neuron centroid. With a small spread, the neuron becomes very selective. With larger spread, distant points have a greater influence and the function approximation will be smoother.

Figure 1 shows a schematic representation of the inverse radial neural model proposed for the estimation of the latex PSD. This model is created using a set

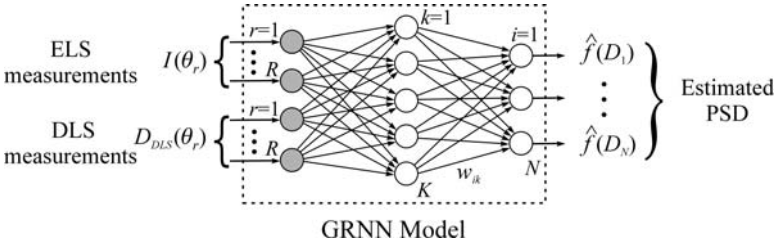


Fig. 1 Inverse GRNN model proposed for estimating the PSD of a latex.

of K discrete PSDs, and their corresponding measurements obtained according to equations 4 and 5. To simplify the problem, the discrete axis D_i and the angles θ_r are assumed fixed, and only the PSD ordinates (f) and the measurement ordinates (I and D_{DLS}) are presented to the model. Each discrete PSD a-priori known lies in the diameter range [50-1100] nm, with ΔD=5 nm. Measurements were taken at the range θ [20-140] degrees, with $\Delta\theta$= 10 degrees. Each input variable $I(\theta_r)$ and $D_{DLS}(\theta_r)$ is represented by R = 13 discrete points, and the total number of inputs to the model is 2*R = 26. The PSDs used for building the model were restricted to be only unimodals and with a fixed log-normal shape, given by:

$$f(D_i) = \frac{1}{D_i \sigma \sqrt{2\pi}} exp\left[-\frac{[ln(D_i/\overline{D}_g)]^2}{2\sigma^2}\right]; \quad i = 1,2,\ldots,N \tag{6}$$

where D_i, $i = 1,2,\ldots,211$ represents the discrete diameter; $\overline{D}_g$ is the geometric mean diameter; and σ is the standard deviation of the PSD.

For generating the learning set, $\overline{D}_g$ was varied in the range [100-1000] nm, at intervals of 5 nm. For each $\overline{D}_g$, 20 distributions were generated, with standard deviations in the range [0.01-0.20] nm, at intervals of 0.01 nm. Hence, 181 different $\overline{D}_g$ values were considered, with 20 PSDs of different standard deviations for each geometric mean, thus yielding a total of K = 3620 learning patterns. All patterns were normalized to fall in the range [0,1]. The network perfectly learned the data, with an approximate root mean square error (RMSE) of 10^{-5}. Note that all the PSDs used during the definition of the GRNN model were simulated on the basis of the same distribution shape; and therefore no outlier was generated.

4 GRNN model validation

Two kind of validations were implemented. First, the GRNN was validated through simulated (or synthetic) examples, since in these cases the solutions are a priori known, and therefore the NN performance can be clearly evaluated. Then, the model was tested through an experimental example that involves a polystyrene (PS) latex of narrow PSD and known nominal diameter. In this case, the true PSD is unknown; but

the best approximation is given by an independent PSD measurement as obtained from transmission electron microscopy (TEM) [24].

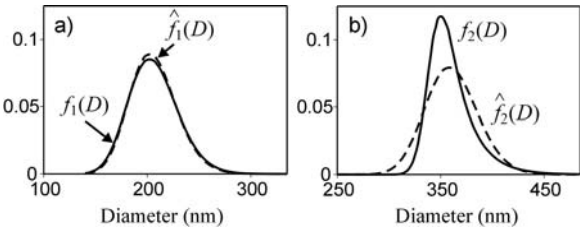


Fig. 2 Two simulation examples for validating the GRNN model: a) a log-normal PSD, $f_1(D)$, and b) an EMG PSD, $f_2(D)$. Comparison with the corresponding GRNN model estimates, $\hat{f}_1(D)$ and $\hat{f}_2(D)$, respectively.

4.1 GRNN model validation with simulated data

Two asymmetric and unimodal PSDs of a PS latex were simulated. The first PSD, $f_1(D)$, follows a log-normal distribution, with $\overline{D}_{g,1} = 205$ nm, and $\sigma_1 = 0.115$ nm. The second PSD, $f_2(D)$, was assumed as an exponentially-modified Gaussian (EMG) distribution, obtained by convoluting a Gaussian distribution (of mean diameter $\overline{D}_{g,2} = 340$ nm and standard deviation $\sigma_2 = 20$ nm), with a decreasing exponential function (of decay constant $\tau = 10$ nm).

The selected "true" PSDs are represented in Figure 2 (in continuous lines). Notice that $f_1(D)$ presents the same shape used for creating the GRNN model, and for this reason it will be useful for evaluating the model interpolation ability. In contrast, $f_2(D)$ exhibits a higher asymmetry than any log-normal distribution, and it was selected to evaluate the ability of the GRNN model for estimating PSDs with different shapes than those used during the model creation. The corresponding estimates are also represented in Figure 2 (in dashed curves). In the case of $f_1(D)$), its estimation is almost perfect. On the contrary, in the case of $f_2(D)$, its estimation is broader than the true PSD; however, the solution is smooth and acceptably close to $f_2(D)$. Additionally, both estimates exhibits only positive values, which is practically impossible to be obtained when traditional regularization routines are used to solve the ill-conditioned inverse problem.

4.2 GRNN model validation with experimental data

A commercial latex standard of PS (from Duke Scientific) of nominal diameter 111 nm was measured through the following independent techniques: 1) DLS; 2) ELS;

and 3) TEM. For the light scattering measurements, a Brookhaven instrument was used. The TEM measurement was obtained after counting about 500 particles, in a Hitachi H-7000 equipment [24].

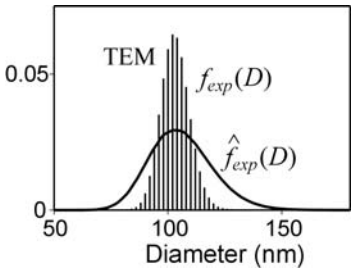


Fig. 3 Experimental example for validating the GRNN model. Comparison of the TEM measurement, $f_{exp}(D)$, with the GRNN model estimate, $\hat{f}_{exp}(D)$.

The PSDs obtained from TEM, $f_{exp}(D)$, is shown in Figure 3 as a histogram, and it is considered as a good approximation of the true (but unknown) PSD. The DLS and ELS measurements were fed into the trained GRNN; and the resulting estimated PSD is indicated as $\hat{f}_{exp}(D)$ in Figure 3. The PSD estimate resulted somewhat broader than the TEM measurement. However, the average diameters of both PSDs are quite similar.

4.3 Error Estimation Indexes

To evaluate the quality of the PSD estimates, the following performance indexes are defined:

$$J_f = \left(\frac{\sum_{i=1}^{N} \left[f(D_i) - \hat{f}(D_i) \right]^2}{\sum_{i=1}^{N} \left[f(D_i) \right]^2} \right)^{0.5} \tag{7}$$

$$E_D = \frac{\overline{D}_n - \hat{\overline{D}}_n}{\overline{D}_n} \times 100 \tag{8}$$

where $\overline{D}_n$ is the number-average diameter of the PSD, that is defined as:

$$\overline{D}_n = \frac{\sum_{i=1}^{N} f(D_i) D_i}{\sum_{i=1}^{N} f(D_i)} \tag{9}$$

Table 1 compares the different performance indexes for the 3 analyzed examples. In all cases, the mean diameters are accurately predicted. The experimental case exhibits the highest J_f index. However, the comparison is against the TEM

measurement, which is narrower than the "true" PSD as a consequence of the limited number of counted particles.

Table 1 Performance indexes for simulated and experimental examples.

	$f_1(D)$	$\hat{f}_1(D)$	$f_2(D)$	$\hat{f}_2(D)$	$f_{exp}(D)$	$\hat{f}_{exp}(D)$
$\overline{D}_n$	206.4	206.3	360.0	360.9	103.2	105.9
E_D (%)	-	0.05	-	-0.25	-	-2.62
J_f	-	0.01	-	0.09	-	0.11

5 Conclusions

A method for estimating the particle size distribution of polymer latexes from combined ELS and DLS measurements was developed. The proposed model utilizes a general regression neural network, that was built on the basis of simulated log-normal PSDs, with particles in a relatively broad diameter range [50-1100] nm. The GRNN model building is straightforward and fast, because no training or validation procedure is required. The proposed approach was successfully evaluated on the basis of both simulated and experimental examples. It was observed that the resulting GRNN was able of accurately recuperating PSDs of log-normal distributions. In principle, asymmetric EMG distributions can be adequately estimated too. Also, the GRNN successfully estimated a narrow PSD of a commercial PS standard, yielding a distribution close to that directly obtained by TEM. From a practical point of view, the neural network constitutes a fast and robust tool, which additionally proved adequate for the resolution of the involved ill-conditioning non-linear inverse problem. With respect to the standard inversion techniques, the network presents the advantages of not requiring any diameter range or numerical inversion method. Also, it has proven to be insensitive to the standard noise measurements. The proposed method has also proven to adequately predict the most difficult case of estimating narrow PSDs.

Finally, the main limitation of the proposed approach is that it was developed for unimodal PSDs. However, the network performance could be extended to more general distributions, by including different PSD shapes during the GRNN model definition. Also, an improved PSD resolution can be attained by reducing the discretization of the diameter axis, and/or by increasing the number of angles at which the measurements are taken. As a future work, a more general tool will be presented with reduced restrictions on the PSD shape.

References

1. Meyer, T., Keurentjes, J.: Polymer Reaction Engineering, an Integrated Approach. In: Handbook of Polymer Reaction Engineering. Chap. 1, pp. 1-15, Eds. Wiley-VCH (2005)
2. Richards, J., Congalidis, J.: Measurement and Control of Polymerization Reactors. In: Handbook of Polymer Reaction Engineering. Chap. 12, pp. 595-678, Eds. Wiley-VCH (2005)
3. Schoenmakers, P., Aarnoutse, P: Chemical Analysis for Polymer Engineers. In: Handbook of Polymer Reaction Engineering. Chap. 20, pp. 1015-1046, Eds. Wiley-VCH (2005)
4. Gao, J., Penlidis, A.: Mathematical Modeling and Computer Simulator/Database for Emulsion Polymerization. Progr. Polym. Sci., 27, pp. 403-535 (2002)
5. Kirsch, A.: An Introduction to the Mathematical Problem of Inverse Problems, Springer-Verlag, New York (1996)
6. Stegmayer, G., Chiotti, O.: Neural Networks applied to wireless communications, In: in IFIP International Federation for Information Processing, Volume 217, Artificial Intelligence in Theory and Practice, ed. M. Bramer, (Boston: Springer), pp. 129-138 (2006)
7. Minari, R. et al.: Industrial SBR Process: Computer Simulation Study for Online Estimation of Steady-State Variables Using Neural Networks, Macromolecular Reaction Engineering, 1(3), pp. 405-412 (2007)
8. Gilbert, R.: Emulsion Polymerization. A Mechanistic Approach, Academic Press, London (1995)
9. Bohren, C., Huffman, D.: Absorption and Scattering of Light by Small Particles, J. Wiley & Sons, New York (1983)
10. Tikhonov, A., Arsenin, V.: Solutions of Ill-posed Problems, Wiley, Washington (1977)
11. Vega, J. et al.: Latex Particle Size Distribution by Dynamic Light Scattering. A Novel Data Processing for Multi-Angle Measurements. J. Coll. and Int. Sci., 261, pp. 74-81 (2003)
12. Vega, J. et al.: A Method for Solving an Inverse Problem with Unknown Parameters from Two Sets of Relative Measurements. Lat. Amer. Appl. Res., 35, pp. 149-154 (2005)
13. Zhao, R. et al.: Application of an Artificial Neural Network in Chromatography - Retention Behavior Prediction and Pattern Recognition. Chem. & Intell. Lab. Syst., 45, pp. 163-170 (1999)
14. Ulanowski, A. et al.: Application of neural networks to the inverse light scattering problem for spheres. Appl. Optics, 37(18), pp. 4027-4033 (1998)
15. Guardani, R., Nascimento, C., Onimaru, R.: Use of Neural Networks in the Analysis of Particle Size Distribution by Laser Diffraction: Tests with Different particle Systems. Powder Tech., 126, pp. 42-50 (2002)
16. Berdnik, V., Loiko, V.: Sizing of Spheroidal and Cylindrical Particles in a Binary Mixture by Measurement of Scattered Light Intensity: Application of Neural Networks. J. Quantit. Spectr. & Radiat. Transfer, 91, pp. 1-10 (2005)
17. Chu, B.: Laser Light Scattering, Academic Press, New York (1991)
18. Scheffold, F. et al.: PCS Particle Sizing in Turbid Suspensions: Scope and Limitations; In: Particle Sizing and Characterization, Eds. T. Provder and J. Texter (2004)
19. Glatter, O. et al.: Interpretation of Elastic Light-Scattering Data in Real Space. J. of Col. and Int. Sci. 105, pp. 577-586 (1985)
20. Vega, J. et al.: Particle Size Distribution by Combined Elastic Light Scattering and Turbidity Measurements. A Novel Method to Estimate the Required Normalization Factor. Part. & Part. Syst. Charact. 20, pp. 361-369 (2003)
21. Gonzalez, V. et al.: Contamination by Larger Particles of Two Almost-Uniform Latexes: Analysis by Combined Dynamic Light Scattering and Turbidimetry. J. of Colloid and Int. Sci., 285(2), pp. 581-589 (2005)
22. Specht, D.: A generalized regression neural network, IEEE Trans. Neural Networks, 2, pp. 568-576 (1991)
23. Wasserman, P.: Advanced methods in neural computing, Van Nostrand Reinhold, New York (1993)
24. Elizalde, O., Leal, G., Leiza, J.: Particle Size Distribution Measurements of Polymeric Dispersions: A Comparative Study, Particle & Particle Systems Characterization, 17(6), pp. 236-243 (2000)

Intelligent Advisory System for Designing Plastics Products

U. Sancin[1] and B. Dolšak[2]

Abstract Plastics product design is very experience dependent process. In spite of various computer tools available on the market, the designer has to rely on personal or supporting experts' knowledge and experience when designing plastics products. Proposed development of the intelligent advisory system presented in this paper involves two methodologies. "Design for X" strategy will be applied to consider specific design aspects for plastic products, while "Knowledge-Based Engineering" will be used for knowledge acquisition, its systematization and utilization within the system. The major benefit of the intelligent support provided by the advisory system will be faster and more reliable product development process, as the system will offer the user some recommendation and advice about material selection and related production process. Thus, the expert team could be contracted. Minimized development process costs along with optimal technical design solutions for plastic products will enable small and medium size enterprises to compete with their plastics products on the global market.

[1] U. Sancin, BSc
Faculty of Mechanical Engineering, University of Maribor, SI-2000 Maribor, Slovenia,
email: urska.sancin@uni-mb.si
[2] Assoc. Prof. Dr. B. Dolšak
Faculty of Mechanical Engineering, University of Maribor, SI-2000 Maribor, Slovenia,
email: dolsak@uni-mb.si

Please use the following format when citing this chapter:

Sancin, U. and Dolšak, B., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 265–274.

1 Introduction

Not long ago, Computer Aided Design (CAD) represented quite revolutionary approach in both, computer and engineering science. Nowadays designers have at their disposal a wide range of CAD applications like drafting, modelling, analysing, simulation, etc. Using CAD tools design is much more effective. Existing CAD is almost perfected on the graphic presentation of design but still has serious limitations at providing recommendation and advice to the designer. Information like material, surface treatment, tolerances, etc. are generally represented as remarks on the drawing or by attaching attributes to a three-dimensional model while only geometric form and dimensions are covered satisfactory.

However, CAD applications are reality in modern engineering design process and working without them is almost impossible to imagine. As mentioned before, the existing computer support does not offer adequate information or advice to designer when dealing with dilemmas about material selection or process selection etc. Therefore designer has to take a decision based on his or her own knowledge and experience. As one person cannot master such extensive range of knowledge to take all the correct engineering decisions, the designer has to collaborate with a group of experts with different expertises.

The main objective of the proposed research presented in this paper is a development of the intelligent advisory system for plastics products design to support designer with advice at product development process. The basic idea is to build and apply a knowledge base about plastics materials, related production processes and corresponding design guidelines. The major scientific challenge behind this goal is related to data mining and knowledge discovery. Knowledge acquisition is expected to be the most complex and time consuming part of the research to build the knowledge base, where domain knowledge and data will be collected, organised and properly encoded.

The intelligent system mentioned here is expected to be a major acquirement for many small and medium sized enterprises, as with its application even single designers with less experience will be able to achieve optimal design solutions. The intelligent system will replace the expert team to a great extent so they will be able to focus on new technology development and knowledge dissemination.

2 Design for X and knowledge-based engineering

In today's consumption oriented society consumers' requests and wishes have to be considered as basic guidelines along with product specification as early as in concept phase of design process. Even successful designers have difficulties when linking all the factors and reaching compromises between them. The methodology called "*Design for X*" (DFX) is extensively applied in engineering practice. "X"

deputizes many quality criteria like appropriateness for manufacturing, assembly, maintenance, service, etc. [1]. Design for manufacturing (DFM) is mostly used in production process by new product development team as it directly refers to manufacturing process and its costs. All members of the development team as well as outside experts need to contribute their part of expertise to enable the effective DFM practice that leads to low manufacturing costs without sacrificing product quality.

DFX methodology needs to be considered in development of the computer aid for supporting specific design aspects, such as design of plastics products. The second but any less important methodology for our research work is Knowledge-Based Engineering (KBE), which is founded on Artificial Intelligence (AI).

AI is a branch of computer science that is concerned with automation of intelligent behaviour [2]. The Artificial Intelligence (AI) applications to design are generally concerned with studying how designers apply human intelligence to design, and with trying to make computer aids to design more knowledgeable. These applications are based on representation of heuristic knowledge (which is less easy to express) as mathematical approach is not appropriate in this case. As mentioned, the part of AI that is particularly concerned with the development of such representations are known as expert systems, or more generally knowledge–based systems, often also intelligent computer systems [3].

KBE is founded on knowledge base, which is existential for intelligent system functionality. KBE is an engineering method in which knowledge about the product, e.g. the techniques used to design, analyse, and manufacture the product, are stored in the knowledge base or product/process model. The model represents the engineering intent behind the geometric design. It contains the attributes of the product such as material type, functional constraints, geometry etc.

Although AI technology is still subject of extensive research and development, many successful AI applications in real–life domains already proved the usefulness of these technologies when dealing with nondeterministic problems that cannot be treated adequately by using conventional approaches, unless the user is possessed of special skills and experience. Engineering design process is certainly one of the domains that very much fit into this scope.

3 State-of-the-art

It is hard to imagine a modern design process without using a computer. In fact, CAD is so extensively applied that in many companies all design work is done using these software tools. Yet, there is a body of opinion that the benefits of applying CAD are below expectations. We believe the reason for this lies in the fact that the existing CAD systems are still not yet adequate as a proper aid to the designer in the design process of a new product. The way in which it is hoped to overcome this bottleneck is to increase the intelligence of CAD systems [4].

Existing CAD systems represent a present state-of-the-art in computer support to design process. Some of the systems were already upgraded with intelligence in some technical professional fields. Significant improvement of reliability and effectiveness in performing various engineering tasks was perceived. AI applications are not only a subject of extensive research and implementation but today's reality. Proceedings of the international scientific conferences "AI in Design", edited by J.S. Gero [5], constitute a good collection of papers related to this area.

Design data are not always well formulated, and almost never complete. Experienced designers can deal with such data reasonably easily, while even the most "intelligent" programs have great difficulties. Designers are also reluctant to assign responsibility for decisions to computer programs, no matter how competent they may appear. One can also argue that encoded design knowledge does not allow designers to express their creative ideas. This is even more important in some specific design domains that have their specific constraints and criteria and therefore require specific approach in design process. For all these reasons, computer (intelligent) support to specific design aspects, including those that are subject of the proposed research, is still quite limited and therefore insufficient.

3.1 Design of plastics products

In today's world, designer should follow quite extensive list of basic steps and procedures to produce a world-class product [6]. Within this process, different design decisions have to be taken, like choosing the appropriate material, production process, tooling, equipment, services, etc. The single designer is not able to reach all the correct decisions so the consultations with experts of different expertise are of high importance. Furthermore, design with polymers requires more involved and upfront engineering approach than ever before. This is why the intelligent computer support to plastics design is essential.

Yet, quite many AI applications have been reported in this particular field of design. In 1996, Rapra Technology Ltd. [7] claimed to lunch first ever knowledge based system for the plastic industry. Most of the later AI applications were addressing separate parts of design process, i.e. the selection of specific materials, such as ceramic [8], or the use of special manufacturing and corresponding tooling processes, where injection moulding is far the most popular [9, 10]. On the other hand, no serious attempt is recorded to develop the intelligent advisory system for supporting plastics products design process as a whole. Therefore, our research presented in this paper represents a novel contribution to this important technical field.

4 Problem presentation

In present time, each product is exposed to the competitive struggle on the market, where success can be expected only for universally optimal design solutions. Consequentially, specific design aspects like design of plastic products are becoming increasingly important and are not left behind in correlation with functionality and economic efficiency. Every competent enterprise with intention to take the leading position in its branch is aware of this fact and is opened for any process application in order to achieve that goal. Therefore, it is no surprise that CAD is so extensively applied.

In spite the fact that modern CAD tools are very strong in graphic presentation, the limitations at providing design recommendations to the user are becoming more and more obstructive. Design projects normally originate in form of problem statement provided to the designer by the company management. These problem statements set a *goal*, some *constraints* within which the goal must be achieved, and some *criteria* by which a successful solution might be recognised. It is usually possible to improve the initial definition of the problem. Yet, many design constraints and criteria still remain unknown and the existing CAD approaches are not able to help designer in dealing with uncertainty and inconsistencies. Thus, the quality of design solution depends mostly on the designer's skill and experience.

Designers face many dilemmas linked with various aspects of the product. Compromises have to be considered at every design step. In order to create as optimal compromises as possible, designers have to possess wide range of knowledge and have to be aware of all influential parameters, or alternatively a team of experts in various fields has to collaborate in development process [11].

The designer often stands at the crossroads as product specifications and customer requirements are very much contradictive with specific design issues like how to produce, to assemble, to maintain or to service the product. In this case, DFX methodologies can be very helpful but often rather not sufficient.

Figure 1 shows product development process, following four phases of design (upper part of the Figure 1): task clarification, conceptual design, embodiment design and detail design.

The designer is progressing through design phases relying on personal and expert team's knowledge. As the alternative to the extensive expert team, which is often not on disposal, we propose a supplement to the existing CAD tools in form of the intelligent advisory system for supporting plastics product design described in this paper. Parallel with research in field of intelligent support to plastics design, the intelligent systems for supporting ergonomic and aesthetic design are also subject of development in our laboratory [12].

It is anticipated the proposed intelligent supporting system to comprise several intelligent modules, each based on the knowledge base for specific design aspect. The lower part of the Figure 1 shows the idea of the proposed intelligent system

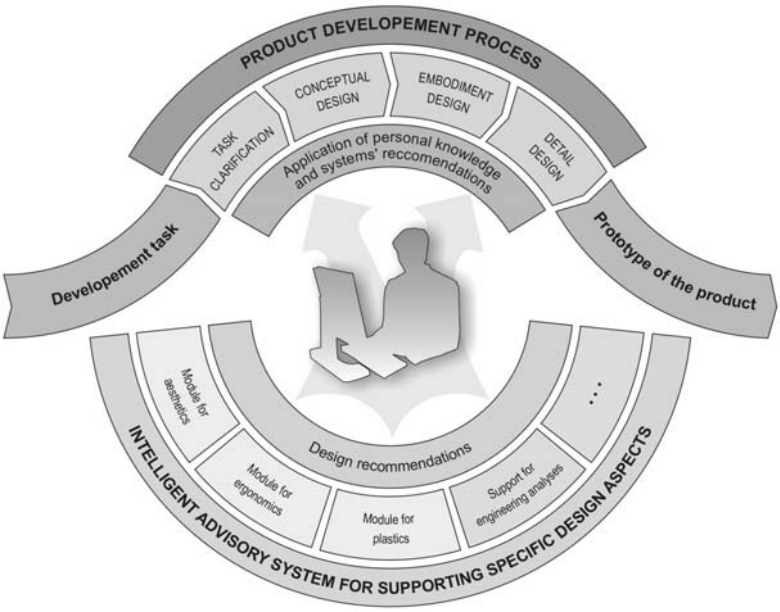


Figure 1 Product development process supported with intelligent advisory system

also presenting the intelligent module for supporting engineering analyses, which was already developed in our research group and gave promising results.

Above all, the main purpose of the intelligent system is to support inexperienced designer by qualified stream of advices in terms of design recommendations and guidelines for specific design aspects. Using this intelligent computer support, the designer will be able to use the "knowledge" encoded in the knowledge base of the system to follow basic steps of design and to find optimal design solutions easier and faster.

5 Intelligent advisory system

To consider the world without plastics is today almost impossible to imagine. Only the most innovative enterprises, which are investing in development and are keeping up with their competition, can be successful. In engineering practise designers usually decide mostly upon well-known tested materials like metals, wood and ceramics. In the present time, this kind of materials can be substituted with others, more suitable for the certain type of product. Plastics are one of those alternative materials, as they can offer optimal characteristics for noticeable lower costs. Due to the assortment of polymers (over 120 thousand) available on the market, the expert working team should be numerous, while planning the new

product. Consequentially, the designer is forced to rely upon knowledge and experience of the working partners.

According to previous writing some major problems of plastics product development process can be outlined and summarised in two basic groups:

1. New materials are not adequately represented in engineering practise due to insufficient knowledge about new materials and related characteristics, as well as due to traditional use of other more familiar materials.
2. Information about plastics, their properties, related machining processes and matching design recommendations are not properly collected and organised for engineering use.

Development of the intelligent advisory system for plastics design will be clearly an acquirement to the great extent, as engineers will finally be able to get some professional recommendation how to deal with some specific design aspects in plastics product design process. In this manner, they will be able to select the material best suited to the purpose of the product without dependency on their limited experience, creativity and product performance requirements.

5.1 KBE and DFX in plastics design

Design is a very complex process. One of the crucial decisions that need to be made within this process is also the selection of the material for the new product. Material selection is always affected by basic demands, like application manner and additional factors, like supplier recommendation, own experience, etc. Moreover, the designer also has to anticipate the production process, semi-product or product assembly, maintenance of the part and environmental component, which becomes especially important due to the pollution of the planet. For all this reasons, DFX methodology has to be considered very thoroughly, especially at plastics product design process. However, designer still can not expect any adequate help in form of recommendation or guidelines when material or technological process has to be selected to achieve maximal quality at minimal costs.

In order to overcome this bottleneck, KBE techniques need to be considered along with DFX methodology, when developing the intelligent advisory system for plastics product design. Figure 2 presents the basic idea and is visually divided

in three components: input, intelligent module and output. Input containing customer's requests, wishes, and technical criteria is general for any design problem, the same as the output with selected material and some design and production guidelines. The intelligent module for plastics product design represents a new component in design process. The preliminary condition for the knowledge-based support to plastics product design process is the adequate knowledge base containing related, well organized DFX knowledge, relations and data.

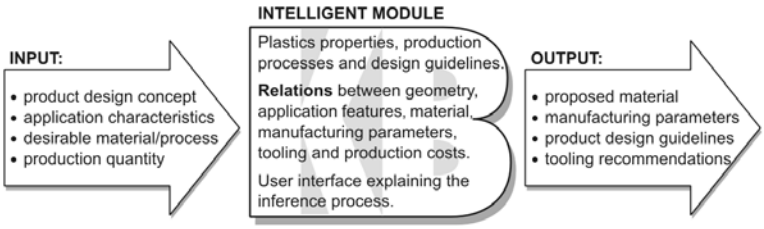


Figure 2 Knowledge-based design of plastics products

It is evident that expert support in decision-making process is essential for almost every designer to perform design of plastics products successfully and efficiently. Thus, we decided to develop an intelligent advisory system to support this important design issue. Above all, the expected development methods include a combination of basic design knowledge with special domain expertise in field of plastics. The knowledge base will contain rules related to selection of the modern plastic materials and correlated manufacturing processes, as well as special guidelines and recommendations for designing plastics products. Different approaches to knowledge acquisition and the appropriate formalisms for the presentation of the acquired knowledge within the computer program will be of special importance. The potential of transparent and modular IF-THEN rules is planned to be compared with more flexible knowledge presentation systems such as fuzzy logic.

Figure 3 shows comparison between conventional approach of plastics product design and design process supported with the intelligent advisory system.

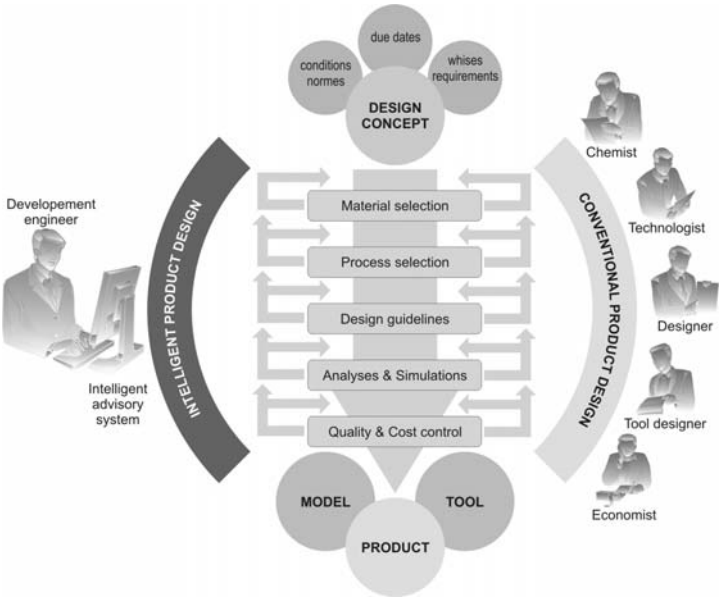


Figure 3 Design process with intelligent system support vs. conventional approach

Requirements, wishes, conditions and due dates are passed to designer either from the management or direct from the customer. Designer's responsibility is to design the model, tool or finished product by carrying out the whole development process. Thus, designer should consult with experts like technologist, chemist, tool designer and economist, who are able to deliver their expert knowledge as this is the only way to achieve performance of optimal solution.

As the alternative to the expert team (Figure 3), some tasks like choosing the material and process, presenting the design guidelines, performing the analyses and monitoring the quality and costs, can be supported by applying the intelligent advisory system proposed here.

In present practice design process is sometimes still successive. The customer provides the designer with input data where the requests frequently predominates technical criteria. Therefore, the designers and technologist are often handicapped when trying to enhance the quality of the product or a process. In such case, the need for intelligent computer support is also very expressive.

The intelligent system will be developed in form of consultative advisory computer tool to be used interactively. The main goal for the system is to apply domain knowledge, relations and experiences from the knowledge base of the system in complex reasoning procedure leading to qualified design recommendations.

In order to enable transparent and efficient system application, the user interface will be developed with a special attention. Regarding the type of input and output data, two different application modes are anticipated. Guided mode (question and answer) will be used mostly at the beginning, when first set of parameters has to be presented to the system. During data processing phase, the system may present additional questions or ask for more parameters. In this case, guided and graphic mode will be used to present the problem to the user. In final phase, the solution will be presented in graphic mode if possible.

6 Conclusions

Knowledge and experience of design experts are of crucial importance for plastics product design process. Thus, young inexperienced designers have many difficulties when facing the challenge to make crucial decisions within complex design process. With development of the proposed intelligent advisory system, knowledge and experiences will be collected, systematised and arranged in the module-based knowledge base.

KBE techniques are already extensively applied in developed world mostly in military, airplane and automotive industry. Small and medium sized enterprises are also aware of KBE advantages but do not have enough human and financial resources for implementation of those techniques in development process. Consequentially, their competitiveness on the market is aggravating as for higher development costs and related product price. The intelligent advisory system for plas-

tics design will help them to achieve their business goal: »maximal quality at minimal costs« less experience-dependent and with higher efficiency. The experts working on one project will be contracted and the team members will be able to dedicate to the new technologies and dissemination of personal knowledge.

The research presented in this paper is a part of the broader research activities, performed by members of our laboratory. The aim of these activities is to develop intelligent advisory systems for supporting product realization process. Our recent results in this research field are two prototypes of the intelligent systems, first to support finite element selection process [13], and the other to support design optimisation considering the results of the structural engineering analysis [14].

The proposed system is also meant to be used in education for the students of engineering and product design as typical representatives of inexperienced designers. In this case, the important feature of the intelligent systems that usually have the ability to explain the interface process will be especially welcome.

References

1. Huang G.Q. (Ed): Design for X - concurrent engineering imperatives. Chapman & Hall (1996).
2. Luger G.F., Stubblefield W.A.: Artificial Intelligence and the Design of Expert Systems. The Benjamin/Cummings Publishing Company Inc, Redwood City (1989).
3. Turban E., Aronson J.E., Liang T.P.: Decision Support Systems and Intelligent Systems. Prentice Hall, 7th edition (2004).
4. Finger S., Ttetsuo T., Martti M.: Knowledge Intensive Computer Aided Design. Kluwer Academic Publishers (2000).
5. Gero J.S. (Ed): Artificial Intelligence in Design '02. Springer (2002).
6. Gordon M.J.: Industrial Design of Plastics Products. John Wiley & Sons (2003).
7. Rapra Technology Ltd.: Knowledge based expert system for the plastics industry. In: Materials & Design. Vol. 17, No. 4, pp. 227 (1996).
8. Sapuan S.M., Jacob M.S.D., Mustapha F., Ismail N.: A prototype knowledge-based system for material selection of ceramic matrix composites of automotive engine components. In: Materials & Design. Vol. 23, pp. 701-708 (2002).
9. Mok C.K., Chin K.S., Hongbo L.: An Internet-based intelligent design system for injection moulds. In: Robotics and Computer-Integrated Manufacturing. Vol. 24, Issue 1, pp. 1-15 (2008).
10. Wang K.K., Zhou J.: A Concurrent-Engineering Approach Toward the Online Adaptive Control of Injection Moulding Process. In: CIRP Annals - Manufacturing Technology. Vol. 49, No. 1, pp. 379-382 (2000).
11. Clarkson J., Eckert C. (Eds): Design process improvement - a review of current practice. Springer (2005).
12. Kaljun J., Dolšak B.: Computer Aided Intelligent Support to Aesthetic and Ergonomic Design. In: WSEAS Transactions on Information Science and Applications. Vol. 2, No. 3, pp. 315-321 (2006).
13. Dolšak B.: Finite element mesh design expert system. In: Knowledge-based systems. Vol. 15, No.5/6, pp. 315-322 (2002).
14. Novak M., Dolšak B.: Intelligent computer-aided structural analysis-based design optimisation. In: WSEAS transactions on information science and applications. Vol. 3, No. 2, pp. 307-314 (2006).

APPLICATIONS 2

Modeling the Spread of Preventable Diseases: Social Culture and Epidemiology

Ahmed Y. Tawfik[1] and Rana R. Farag[2]

Abstract This paper uses multiagent simulation to examine the effect of various awareness interventions on the spread of preventable diseases in a society. The work deals with the interplay between knowledge diffusion and the spreading of these preventable infections in the population. The knowledge diffusion model combines information acquisition through education, personal experiences, and the spreading of information through a scale-free social network. A conditional probability model is used to model the interdependence between the risk of infection and the level of health awareness acquired. The model is applied to study the spread of HIV/Aids, malaria, and tuberculosis in the South African province Limpopo. The simulation results show that the effect of various awareness interventions can be very different and that a concerted effort to spread health awareness through various channels is more likely to control the spread of these preventable infections in a reasonable time.

1 Introduction

Social simulation models have been used to study knowledge diffusion through a social network [2] as well as modeling the spread of epidemics [3]. These models, unlike techniques that focus on cross sectional statistical analysis [8], also capture the factors affecting the individual and at same time allow global trends to emerge in the simulation environment. The bridging of the gap between the micro and macro levels is a useful feature in the study of knowledge diffusion and epidemics. Moreover, hypothetical scenarios are more easily evaluated using social simulations than other epidemiological models. In addition, the relative simplicity of the

[1] Ahmed Y. Tawfik
The University of Windsor, Windsor, ON N9B 3P4, Canada, atawfik@uwindsor.ca
[2] Rana R. Farag
The German University in Cairo, New Cairo, Egypt, rana.hassanfarag@student.guc.edu.eg

Please use the following format when citing this chapter:

Tawfik, A.Y. and Farag, R.R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 277286.

"agent" in a social simulation makes it possible to explore complex interactions over substantial time durations that are difficult to capture in purely mathematical models.

One such interaction is the interaction between knowledge about disease prevention and the effectiveness of various awareness spreading media in controlling the spread of a preventable disease. In this context, a disease is considered preventable if immunization, life style changes, and other means of prevention are effective in controlling the spread of the disease.

Knowledge about disease prevention may be acquired through formal education, personal experiences, and through advice from a social network of friends, relatives, and neighbors. A physician or a licensed health practitioner would have acquired good knowledge in the area of disease prevention through education. In fact, there is a continuum representing the validity of preventive health knowledge acquired through education. At one end of this continuum, we find the physicians and health care professionals, and on the other end we find individuals with no formal schooling.

Disease prevention knowledge acquired through personal experiences is typically unreliable especially for diseases with long incubation periods like HIV/Aids, tuberculosis, or malaria as individuals are not likely to correlate risky behavior to a latent effect. However, personal experiences could guide an individual trying to avoid mosquito bites. By trying insect repellent or mosquito net, the individual may acquire knowledge that helps in preventing malaria.

Mass media outlets like newspapers, radio, and television are considered useful in promoting healthy life styles and combating diseases. However, the effectiveness of mass media depends on their degree of penetration, the consistency of the message, and the degree of trust that the individuals accords to the media outlets.

Word-of-mouth advice from friends and relatives is also an important means for knowledge diffusion. However, the quality of the information usually deteriorates as it spreads, particularly among the less educated groups.

This paper presents a social simulation model that can be used to assess the effectiveness of various awareness interventions. Section 2 presents the elements of the social simulation model including the agents, the social networks, and the structure of the rules governing the spread of information. Section 3 introduces the simulated society in Limpopo, South Africa and the specific parameters and rules used in the simulations. Section 4 examines the effect of various health awareness interventions on the spread of the diseases and the size of the population. Section 5 is the conclusion of this work.

2 The Social Simulation Model

Initially the model starts by generating a set of agents that represent the demographics of the society to be simulated. The agents are grouped into households according to a set of predefined criteria. If agents in the household share knowledge the health awareness level becomes the same for all agents in the household.

A social network links households based on a set of criteria. The network includes special highly connected nodes representing influential individuals. Knowledge spreads along the social network. Based on their educational background agents may acquire knowledge from other sources including formal education and newspapers. Agents also acquire information from radio and television. As the simulation progresses, children grow up and new households are formed while other households dissolve as agents die. Children in a dissolved household are adopted by other households in the social network. The probability of infection of agents at any time depends on age, gender, and their awareness level.

2.1 The Agents

Each agent has a health status that represents whether the agent is healthy, susceptible, infected and incubating, infected and symptomatic, recovering, immune, or dead. Each health state, except dead, lasts for a period of time that depends on the infection and age. Agents may suffer from more than one infection at the same time. There are some interactions between concurrent infections. For example, any infection is more severe if the patient has HIV.

The agent also has an education level that evolves for children according to the population's demographic data. Education affects awareness in more than one way; it determines the awareness obtained from formal education and the reliability of the agent is relaying information.

Agents also have other demographic characteristics including gender, race, age, lifestyle parameters (e.g. number of sex partners), and a variable representing the degree of religiosity. The degree of religiosity determines the influence of information from preachers on the individual. Preachers are members of a group of highly influential agents that also includes physicians, other health care workers.

2.2 The Household

A household is formed as a result of a marriage. Members of the same household share the same level of health awareness. Children are born in a household and every agent is a member of a household. Unmarried adult children continue to be members of the household. A household dissolves if all adult member of the household die. Any children in the dissolving household are adopted by other households. The ability of a household to accommodate orphans depends on the health state of its members.

The household is the unit of membership in the social network. Media exposure is assumed to be the same for members of the same household. Radio and television ownership is assigned per household. Household members also share income; however, the current version of the simulation model does not take in consideration these economic factors. The interaction between household income and health has been studied previously [1]. Other household attributes include its location as it affects the social network, which typically includes some neighbors.

2.3 The Social Network

The social network is a scale free network where most nodes representing ordinary households with a small number of connections to other households. However, there are some high degree nodes representing influential agent such as the physicians, health care workers, and preachers. Fig. 1 shows a social network where ordinary households are shown in blue while more influential nodes are marked by a box around them.

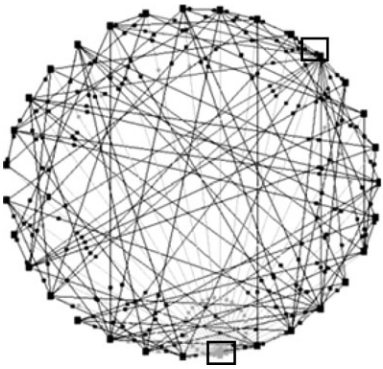


Figure 1 The social network

The network is dynamic to accommodate changes such as household creation and dissolution. There are some compatibility criteria to guide the creation of links between households including educational background and household location.

2.4 Acquisition and Spread of Information

While personal experiences are excellent source of knowledge [4], they are less relevant in this application due to the long incubation periods as discussed earlier. Other sources of knowledge are formal education, peers in the social networks, and influential agents in the social networks.

Each agent is assigned an initial awareness level based on education. Table I provides the interpretation of awareness levels. The higher the level of awareness, the more informed the agent is. Level 9 is assigned to physicians and Level 8 is assigned to other health care workers based on their formal education. The education attainment of a child is determined based on the education attainment of adults in the household.

The increase in awareness level from media differs according to the media and the frequency of the agent's exposure to it. Such that, daily television watching, radio listening, or weekly newspaper reading increases the awareness of the agent by a fixed amount monthly.

Every month the social links of each household are visited and knowledge is shared with a fixed probability assuming that individuals share preventive health

information occasionally. If two friends share knowledge, then their relative awareness is readjusted as follows:

Considering the awareness A_u and A_v of agents u and v, respectively, whenever $A_u > A_v$ then the readjustment is done as: $A'_v = A_v + \gamma A_u$, *and* $A'_u = A_u - \beta A_v$, where $1 > \gamma > \beta > 0$ to reflect that knowledge diffusion is partial and that the correct information is typically more convincing than incorrect information.

Table 1. Awareness Levels

Level	Interpretation
0	False information
1	
2	
3	*Awareness*
4	*Increases*
5	
6	
7	
8	
9	Correct information

The awareness levels of the physicians and health care workers are not affected because they are confident of their knowledge while the awareness of ordinary individuals is increased as a result of the interaction with them. The amount of the increase is proportional to the agent's education.

2.5 Spread and Effects of Epidemics

The spread of diseases is based on a probability model that takes into account typical epidemiological factors such as age, gender, and lifestyle. However, we add a factor that represents the effect of awareness. For example, the rate of occurrence of infections can be presented as a proportional hazard model [5], such that the factors affecting the hazard rate $h(t)$ include both epidemiological and awareness factors represented by the vector $X = [x_0, x_1, \ldots, x_n]$ weighed by the coefficients $\alpha_0, \alpha_1, \ldots, \alpha_n$, *or* $h(t) = h_0(t)\, e^{\sum \alpha_i x_i}$ where $h_0(t)$ represents the unconditional hazard.

Naturally, the coefficient for awareness should ensure that the probability of infection decreases as the awareness increases. Of course, the choice of the survival model and its coefficients will have a great impact on the results.

Once infected, an agent develops symptoms after a random incubation period that depends on the disease, and subsequently may recover or die after another time period. Agents who recover or do not get infected will die at an older age as a result of other causes. A secondary infection to an agent suffering from HIV acts as a time accelerator and the agent die earlier as a result.

3. Case Study: Limpopo

As a case study, we have simulated a village in Limpopo Province in South Africa, which is a well studied example [1] of a community facing great difficulties because of preventable diseases. These problems include high levels of infections with HIV/AIDS, tuberculosis, and malaria. As in other regions in sub-Saharan Africa, the spread of these preventable diseases has had a significant impact on the life expectancy, and on the regional economy.

3.1 The Population

The model starts with 500 individuals whose demographic profile is consistent with recent census data[3] with respect to gender and age. The percentage of the married couples in the actual Limpopo society, which is approximately 22% of adults aged 15 and over, determines the number of households. Throughout the simulation, 1.08% of men marry each year according to official recorded marriage statistics. To account for common law relationships, the model considers that 2.2% of men marry per year, and hence the same number of households constructed. Despite the fact that, in Limpopo's real community, more than one couple may live in the same household, due to the expenses of constructing a new one, the model considers each couple to be living in a separate household so that their friendship relations with other households can be separate. Matching a couple is done randomly, but for a couple to marry, they should meet the following criteria:

- Both partners should be over 15 years old.
- The husband is not older by more than 10 years and the wife is not older with more than 5 years.
- They should be of the same race with a probability of 90%.
- The husband's education level can be higher than the wife's with at most three levels or the wife's education level can be higher with at most one level.

While constructing the community, the model assigns each couple a random number of children between 0 and 3 of each gender. Children inherit the ethnicity of the parents. Each year, a percentage of 4.2% are born (average of birth rates from 1999 to 2003) and the model randomly chooses households for new babies. Single adults in the initial population are also assigned randomly to households.

3.2 Sources of Knowledge

Education levels and initial health awareness is also assigned to individuals based on available statistics. Table 2 shows the statistics[4] used. Health awareness levels within the ranges specified in Table 2 are assigned to agents randomly.

---

[3] Statistics South Africa, Stats in Brief, 2006.

[4] UNESCO, Estimates and projections of adult illiteracy for population aged 15 years and above, by country and by gender 1970-2015.

As the number of physicians in Limpopo is one for every 11,000 individuals and the ratio of nurses to physician is 5.3 to 1, our initial population of 500 is assigned a part time physician who can only see 30 patients a month and a nursing staff that can see 159 patients each month. Priority is given to symptomatic patients but they also see others.

The model assumes that 70% of the population is somewhat religious but the influence of the preacher on a particular agent is proportional to the religiosity factor of the agent**.**

Table 2. Education and health awareness in the population

Education	Health Awareness Level	Percentage
No schooling	0-1	33.4%
Some primary	0-2	14.1%
Completed primary	2-4	5.5%
Some secondary	2-5	26.1%
Completed secondary	3-6	14%
Higher education	5-8	6.8%

According to a recent study[5], the percentage of the South African population that listens to the radio only once a week is 12.8% and the percentage that listens daily to the radio is 79.3%. While the percentage that watches the television once a week is 10.7%, and daily television watchers constitute 67.3% of the population. Moreover, 40.6% of the population read newspapers on weekly basis. These figures were used to incorporate the effect of media in the simulations.

3.3 The Epidemics

HIV/AIDS primarily infects the 15 to 49 age group. The infection rate is set to 21% which is the average between the World Health Organization statistics and UNAIDS statistics. Mother-to-child infection during pregnancy, birth and breast-feeding occurs with a probability of 25% and children infected through this process do not survive more than 4 years.

Throughout the simulation, the model only considered HIV transmission through either sexual contacts or through mother-to-child transmission. A recent study indicated that the rate of male-to-female infection in South Africa is as high as 74% to 100% [7], and UNAIDS[6] estimates that male-to-female transmission during sexual contact is about twice as likely to occur as female-to-male ones. Therefore, the model considers that with each assigned contact if the male is HIV-

[5] BBC World Service Trust, Research Summary Report: African Media Development Initiative, 2006.

[6] UNAIDS, Women and AIDS Fact Sheet, 2004.

positive then with a probability of 87% the female may become infected too but if the female is HIV-positive, the male may get infected with a probability of 44%.

In the initial community, the model chooses 0.511% of the agents randomly to be infected with tuberculosis. During the simulation, 0.6% of the population gets infected annually[7] According to available WHO statistics, 0.18% of the population that gets infected with TB is HIV-positive adults, 0.12% is normal adults, and 0.3% is children, since they are more liable to get infected. Also, 7.2% of infected people die annually, and 40% are cured.

Children under 4 who are co-infected with HIV/AIDS and TB die. Older agents also die from the co-infection if they have reached the symptomatic phase of HIV characterized by a very weak immune system.

In Limpopo, 6369 individuals reported Malaria infection in 2006 which is equal to 0.11% of Limpopo's population[8], but still most of the cases go unreported, so the model assumes that 1% of the agents get infected. Children under 5 years represent 60% of the infected population, the remaining 40% is equally divided between school aged children and adults.

Children under 6 years who are co-infected with HIV and malaria die, while 40% of infected children in this age group die as a result of a malaria infection alone. Malaria also claims the lives of 20% of infected children in the 6 to 14 age group and 10% of adults who are not in the symptomatic phase of HIV. It also causes a precipitous death for symptomatic HIV/AIDS patients.

4. Effect of Awareness Interventions

To study the effects of various awareness interventions, we simulated the system described in Section 2 and the case study introduced in Section 3 using Repast [6]. As we could not obtain enough data to properly validate the hazard model and its parameters, the results presented in this section should be considered illustrative of what social simulation models could produce. Also, in these simulations, we did not take into account changes resulting from medical advances, or changes in the education attainment over the simulation period. Under the assumed set of parameters, the simulations evaluate the following scenarios over 100 year period:

- Scenario 1: Base scenario with no awareness intervention such that an agent's awareness is solely determined by the agent's education.
- Scenario 2: Agents use their social network to share knowledge but did not have access to advice from medical professionals nor preachers.
- Scenario 3: Agents get knowledge through mass media outlets only.
- Scenario 4: Medical professionals and preachers are the only ones spreading knowledge through the social network.
- Scenario 5: Agents share knowledge within each household only.

[7] WHO, Global tuberculosis control: surveillance, planning, financing: WHO report 2007.

[8] Department of Health: South Africa, Malaria cases in South Africa: 1999-2007.

- Scenario 6: All channels for knowledge sharing listed above are enabled.

In all the above scenarios, we recorded the population growth and the number of agents infected by each of the three epidemics. Each scenario was run several times for cross validation and the average result reported here. As expected, the population in Scenario 6 was healthier than all the others and grew from 500 to 8500 in 100 years. In Scenario 6, HIV/AIDS was eliminated within the first 32 years while tuberculosis and malaria infections went down as well.

Scenario 2 was the least effective intervention as HIV/AIDS infected more than 40% of the adult population. However, the total population grew to 2230 over 100 years. These results suggest that the dilution of knowledge through the social network in the absence of reliable sources does not add to the population's awareness. These results are obtained using the information transfer rate for correct information (γ) as 0.3 and that of incorrect information (β) as 0.3.

Table 3 gives the population at the end of 100 years and the percentage of the population infected with HIV/AIDS. It is clear that the population grows as the mortality rate due to HIV decreases and the death rate because of these infections remained low.

Table 3. Population and HIV infection at the end of the simulation period

Awareness Intervention Scenario	Final Population	HIV/AIDS % of adults	Tuberculosis (%)	Malaria (%)
1. Formal Education	2204	41.5%	0.55%	0.86%
2. Education + Social Network	2229	41.4%	0.54%	0.86%
3. Education + Mass media	3801	25.5%	0.44%	0.76%
4. Education + Reliable community sources	5161	3%	.0.31%	0.63%
5. Education + Household	7209	0% *	0.17%	0.25%
6. All awareness interventions	8500	0% **	0.11%	0.19%

* Reached after 65 years ** Reached after 32 years

The fifth scenario gave the best result for a scenario in which only one method is enabled, due to the presence of many adults living in the same household, so if the awareness level of each became the highest of all their awareness levels, because they live together and have much time to convince each other with what they know, a lot of agents will have high awareness, and therefore the information will be disseminated more quickly.

The results highlight the role of information sharing within the household and in the community. However, these results also warn against the spread of misinformation throughout the social network.

5. Conclusions and Future Research

This study confirms that social simulations can be a useful tool to study the effects of awareness interventions. It shows that various interventions can have significantly different results. As such, this work illustrates an application of multi-agent systems to an important problem. It also shows how to combine traditional epidemiological models and social simulation models to study and analyze the spreading of preventable diseases. The validity of these traditional epidemiological models remains of course the domain of epidemiological studies.

It is important to note that the parameters used in the case study relied on available statistics. However, in the simulations, many additional parameters were arbitrarily chosen. To assess how the results are affected by these parameters, we would like to perform some sensitivity analyses.

Acknowledgments The authors would like to thank Statistics South Africa for providing useful statistical data. The first author acknowledges the support of the Natural Sciences and Engineering Research Council (NSERC), the University of Windsor International Development Research, Education, and Training Fund (IDRET). Both authors thank The German University in Cairo for making their facilities available for this research.

References

1. Alam, S.J., Meyer, R., and Zeirvogel, G. Modelling the socio-economic impact of HIV/AIDS in South Africa. In WCSS 2006: First World Congress on Social Simulation, Kyoto, Japan, August 21-25, 2006.
2. Cointet, J.-P. and Roth, C., How realistic should knowledge diffusion models be? Journal of Artificial Societies and Social Simulation, Vol. 10, No. 3, 2007.
3. Huang, C.-H., Sun, C.-T., Hsieh, J.-L., and Lin, H., Simulating SARS: Small-world epidemiological modeling and public health policy assessments, Journal of Artificial Societies and Social Simulation, Vol. 7, No. 4, 2004.
4. Kobti, Z., Snowdon, A.W., Rahaman, S., Dunlop, T., and Kent, R.D. A cultural algorithm to guide driver learning in applying child vehicle safety restraint. In the 2006 Congress on Evolutionary Computation, pp. 1111-1118. IEEE, July 2006.
5. Leemis, Lawrence M., Reliability - Probabilistic Models and Statistical Methods, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1995.
6. North, M.J., N.T. Collier, and J.R. Vos, Experiences Creating Three Implementations of the Repast Agent Modeling Toolkit, ACM Transactions on Modeling and Computer Simulation, Vol. 16, Issue 1, pp. 1-25, ACM, New York, New York, USA, 2006.
7. Pettifor, A., Hudgens, M., Levandowski, B., Rees, H., and Cohen, M., Highly efficient HIV transmission to young women in South Africa., AIDS, Vol. 21, No. 7, pp. 861-865, 2007.
8. Poundstone, K.E., Strathdee, S.A., and Celentano, D.D., The Social Epidemiology of Human Immunodeficiency Virus/Acquired Immunodeficiency Syndrome, Epidemiology Review, Vol. 26, pp. 22-35, Oxford University Press, 2004.

An Intelligent Decision Support System for the Prompt Diagnosis of Malaria and Typhoid Fever in the Malaria Belt of Africa

A. B. Adehor[1] and P. R. Burrell[2]

Abstract Malaria is endemic in Africa, though curable it is difficult to manage the prompt diagnosis of the disease because available diagnostic tools are affected by the harsh tropical weather. Also, the lack of electricity for the storage of current diagnostic tool in the rural areas as well as the fact that it has signs and symptoms that are similar to those of typhoid fever; a common disease in the region as well, is a major setback. This paper describes the research and development in implementing an Intelligent Decision Support System for the diagnosis of malaria and typhoid fever in the malaria subregions of Africa. The system will be mounted on a laptop, the one child per laptop, which will be powered by a wind-up crank or solar panel. The region chosen for our study was the Western Subregional network of malaria in Africa.

1. Introduction

Malaria is a climate sensitive disease and the parasites that transmit the disease strive very well in Africa's tropical region. Although the disease is curable, it is estimated that a child is killed every 30 seconds and there is an annual report of 500 million cases in Africa [1], [2]. In Africa, there are four Subregional networks (they are: Central Africa, East Africa, Southern Africa and West Africa) set up by the Roll Back Malaria Partnership to combat the disease from different fronts [3].

Although the disease can be managed at home by people with a minimum of 4-5 years education [4], [5]; its prompt diagnosis is hindered by the fact that current diagnostic tools are affected by the harsh tropical weather; the lack of qualified medical laboratory technicians to read test results; the lack of regular or no supply of electricity to preserve available diagnostic tools and the lack of adequate

[1] Mr. A. B. Adehor
London South Bank University, SE1 0AA, UK email:adehor@aol.com

[2] Prof. P. R. Burrell
London South Bank University, SE1 0AA, UK email:phillb@lsbu.ac.uk

Please use the following format when citing this chapter:

Adehor, A.B. and Burrell, P.R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 287296.

transport means to transport patients from the rural areas to the urban areas. These all contribute to the set back in the fight against malaria. Above all, the lack of basic social amenities in the rural areas prevents qualified medical personnel from taking up assignments in these areas of Africa. Thus, the treatment of the disease is left to resident healthcare personnel.

The fact that malaria can be managed at home by people with 4-5 years education does not make it a simple case. The complexity of the management of the disease is attributed to the fact that other fibrille illnesses have signs and symptoms that are very similar to those presented by malaria patients. One such disease is the water related disease Typhoid fever, which is also common in this region of Africa. Typhoid fever is caused and transmitted as a result of poor hygiene. It is known that a child dies every 15 seconds from water related disease [6], of which typhoid fever is one.

Thus, based on the fact that malaria has signs and symptoms that are similar to those of other fibrille diseases, makes home management, as well as management of the disease by healthcare personnel, difficult. This paper, describes the design and development of an Intelligent Decision Support System (IDSS) to aid in the management of the disease both at home and healthcare centres in rural areas. The system is intended to operate as a stand-alone system mounted on a desktop or laptop computer which will be powered by a solar panel or wind-up crank [7]. The system is also intended to be operated by people with little training.

2. Method

Our study was based on the Niger-delta region of Nigeria, in the West African Subregional networks [3], where malaria and typhoid fever are known to be prevalent [8].

The research was approached from two perspectives, which are: 1. whether the diseases could be diagnosed based on signs and symptoms in the region, 2. what people of the region do when they have fever.

The essence of the adopted method was to ascertain if both diseases could be diagnosed differentially based on signs and symptoms by following the principle that medicine is evidence-based and to provide possible applications in the wider sense (i.e. Africa and other malaria endemic region of the world).

3. Survey Findings

We carried out surveys in two regional states (i.e. Delta and Rivers States respectively) of the Niger-delta region of Nigeria in West Africa. In the first survey, 70 questionnaires returned by physicians were used to ascertain whether they could actually diagnose malaria and typhoid fever differentially based on signs and symptoms. This indicated that 60.4% of the physicians agreed to this fact. This finding was further substantiated by interviewing three consultants from the University of Portharcourt Teaching Hospital Portharcourt River state and the Federal Medical Centre Asaba Delta state Nigeria respectively.

A second survey targeted demography, in other to ascertain the demographic attitude when people have fever. In total, 330 questionnaires returned show that 41.4% actually go to a pharmacist to explain their condition; 36.2% will take medication based on previous prescription; 6.9% buy drugs without prescription from a chemist and 15.5% apply traditional African medicine.

3.1 Syndromic Diagnosis of Malaria

Our initial survey findings from physicians and the interview sessions with the medical consultants confirm that malaria can be diagnosed based on signs and symptoms, which is in accordance with the research work carried out by Bojang et al [9] on the syndromic diagnosis of malaria table 1.

Table 1. "Sensitivity and specificity of different methods of diagnosing malaria in Gambian children during high malaria transmission season"[3]

Method of diagnosing malaria	Malaria diagnosed if score >=	Sen.(%)	Spec.(%)	PPV
Field worker using algorithm.	7	88%	62%	55%
	8	70%	77%	62%
Computer calculated score using algorithm.	7	89%	63%	56%
	8	70%	78%	63%
Computer calculated score using a simple count of signs and symptoms.	4	90%	38%	43%
	5	73%	59%	48%
	6	46%	81%	56%
Physician's diagnosis without laboratory results.		82%	61%	53%
Physician's diagnosis after seeing laboratory results.		100%	71%	65%

The results from the work of Bojang et al [9] show that a count of 7 to 8 signs or symptoms, by field workers using algorithms and computer generated algorithms, gave a high positive predictive value (PPV), while the PPV of diagnosis by physicians showed these results increased. The only problem that affected the results of these findings is that field workers often enter wrong signs or symptom, thus affecting the PPV of the disease.

However, such problems associated with entering wrong signs or symptoms are not eminent in the IDSS; as this system applies the traditional method of evidence-based medicine. The system asks questions in two formats: 1. Questions that are directed at users, requires the user to critically observe the patient for specific signs or symptoms depending on the question generated from previous answer to a

[3] Keys on the table: Sen. = sensitivity, spec. = specificity and PPV= positive predictive value"

question, 2. The second question is directed at the patient. This prevents users of this system from entering wrong signs or symptoms. The system also does not require users to observe patient for signs of splenomegaly or hepatomegaly based on Bojang et al [9] findings.

3.2 Differential diagnosis of Malaria, Typhoid fever and other Fibrille Illnesses

The syndronmic diagnosis of malaria, based on the aforementioned survey findings would suggest that the disease can be diagnosed differentially from other fibrille diseases based solely on signs and symptoms. We used a simple model to capture signs and symptoms that are similar to known fabrille diseases in the region from the consultants interviewed.

The model is a simplistic differential diagnostic model for the diagnosis of malaria, typhoid fever and unknown-fever, in which individual modules with signs and symptoms can be encapsulated [10], and methods are used to access each module (figure 1). Questions are generated as the user interacts with the system. At this stage, unknown-fever could be meningitis, pneumonia or pyrexia.

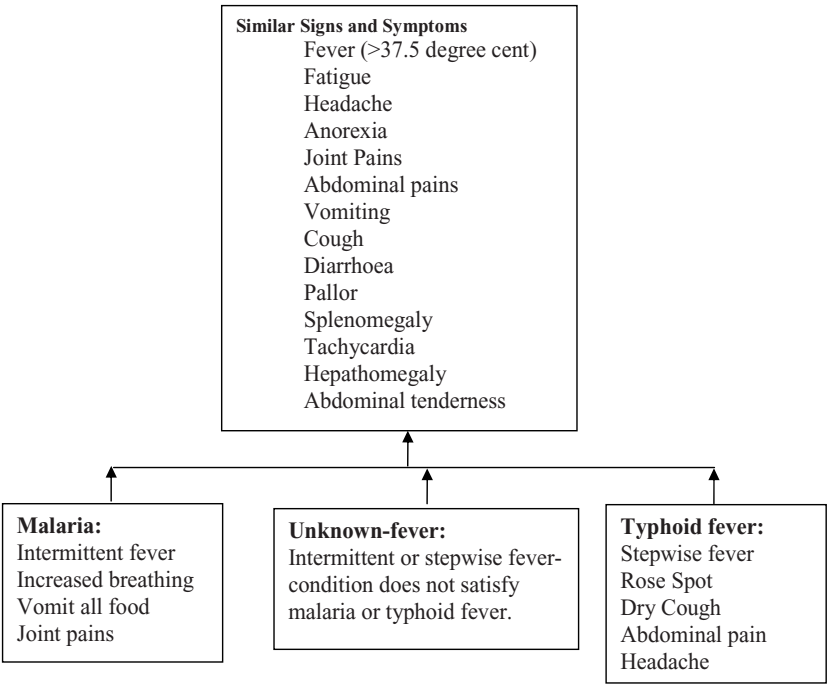


Figure 1 Simple Differential Diagnostic Model for Diagnosing malaria, typhoid and unknown-fever

4. Knowledge Analysis and Representation

The knowledge analysis of the system was carried out using the Mockler Situation Analysis methodology [11]. The result of our situation analysis indicated that there were 8 building blocks (figure 2) upon which the foundation of the differential diagnosis of malaria, typhoid fever and other fibrille diseases could be based. This was aided by interviewing five physicians in our selected region in the West African Subregional network.

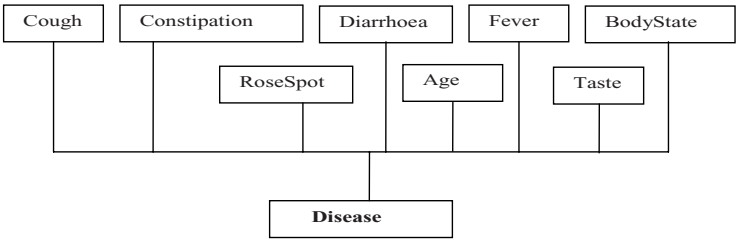


Figure 2 Building Blocks for the Differential Diagnosis of fibrille Diseases.

Thus, in order to effectively represent our findings; we represented each building block (i.e. cough, constipation, rosespot, diarrhoea, fever, bodystate, taste and age) in a decision table as rules. The disease block has rules of its own which are encapsulated [10] and each of the 8 blocks can only gain access to the various signs or symptoms through questions directed at the user or the patient. The decision tables were passed on to the consultants at the University Teaching Hospital Portharcourt to incorporate uncertainty in the line of reasoning. In order to have a reference standard for the signs and symptoms on each table, as compared to the system generated certainty factor (CNF), the experience of both consultants was incorporated into the system (CNF in a scale of 0-100). The system will diagnose the illness based on the answers provided by the user, as depicted in figure 3.

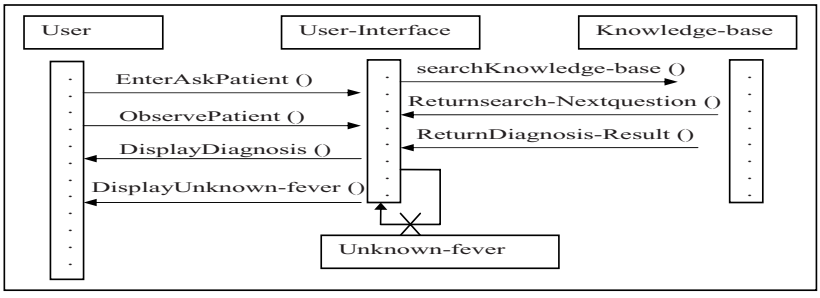


Figure 3 Simple component Interaction of the system with User

4.1 System Design and Implementation

The system was designed and developed using rapid prototyping with a simple expert system shell because of its simplicity and fast learning curve. The knowledge-base of the shell holds details of the heuristics as shown in the general architecture of the system (figure 4).

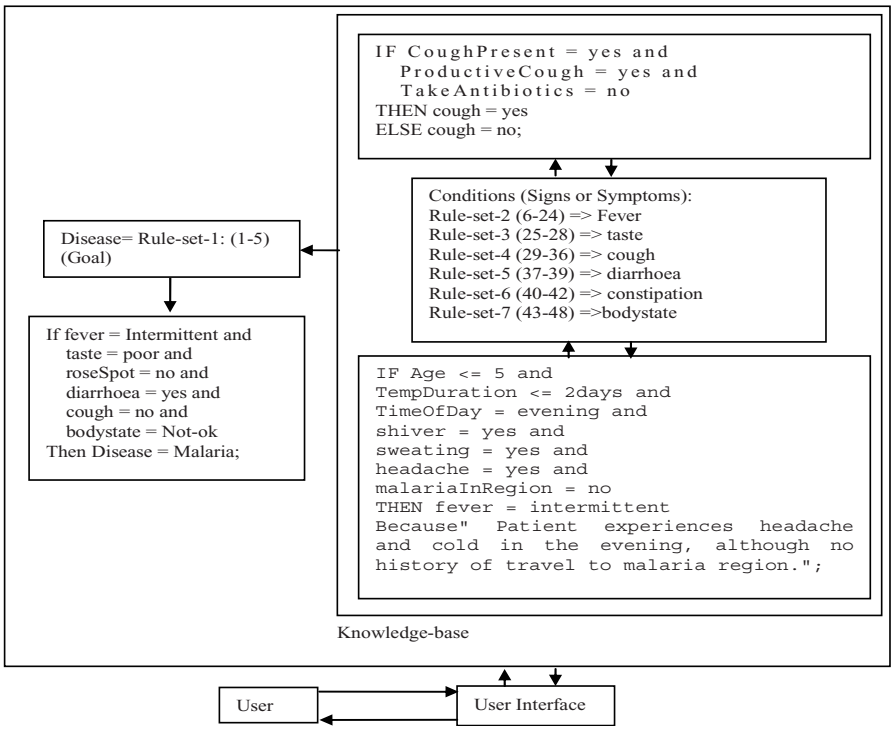


Figure 4 General System Architecture

The decision table for each building block has a group of questions that are asked in such a way that a set of three questions or more need to be answered in order to prove that a patient, suspected of having cough, actually had cough and that the cough is not as a result of any medication. The knowledge-base responds to each question by searching and generating the next question in accordance to the user and patient answers. This same principle is applied in all the other building blocks. For example, to prove that a patient has fever, a total of 18 questions will be asked in different combination and the question combination depends on answers provided by the user.

Thus, the system works in such a way that questions asked are relevant to a particular hypothesis [12].

The system has a total of 53 rules in its knowledge-base; of which 5 depicts the disease-state (i.e. 2 malaria, 2 typhoid fever and unknown-fever) and the

remaining 48 rules represent the building blocks (fig. 2). Thus, for a particular sign or symptom to be confirmed as being present in the patient; each set of questions relating to a hypothesis is proved to be true and all sets, that have been proved to be true, then combine with other confirmed signs or symptoms to give the final disease diagnosis.

The system can also give explanations as to why a particular question was asked as well as to how it arrived at the diagnosis of the disease and how certain it is regarding the diagnosis. Figure 5 shows how the system prevents users from entering incorrect signs or symptoms.

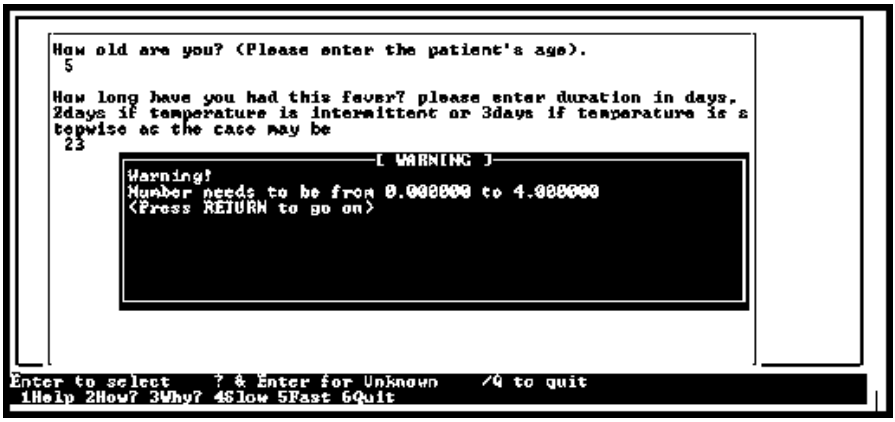


Figure 5 User prevented from entering incorrect data or signs and symptoms.

5. System Evaluation

So many diagnostic tools have been developed over the years for the prompt diagnosis of malaria and typhoid fever and many more are still being sort after. The problem is not that these tools lack the efficacy of diagnosing these diseases, but the lack of qualified medical personnel to actually interpret the test results. Also, some test like the extraction of bone-marrow for detection of typhoid fever bacteria, with an accuracy of 90%, is very painful and there is no less painful or simpler way of extracting the typhoid bacteria. However, of the known tests for the early detection of typhoid fever; the Polymerase Chain Reaction test proved to be very effective but it is affected by the harsh tropical climate as well as its high implementation cost [13].

The sure test for the diagnosis of malaria is the thin and thick blood smear. This test, though effective, lacks qualified people in rural areas to read smear results. The lack of a constant electricity supply in the rural areas is another big hindrance as microbiological chemicals require cold storage medium.

However, research work by Bojang et al [9], shows that malaria could be diagnosed based on signs and symptoms and that one does not have to detect splenomegaly or hepatomegaly in a patient in order to diagnose malaria (table 1).

Other authors like Chandramohan et al [14], have written algorithm for diagnosing malaria. Research studies [15], show that clinical diagnosis based on signs and symptoms is also justifiable.

Other work carried out in the area of malaria diagnosis, utilising different forms of information systems, can differentiate malarial specie from blood smear [16], use signs and symptoms [17] and use ontology driven multi-agents [18]. These systems have demonstrated the effectiveness of such methods, but they rely heavily upon an established clinical and IT infrastructure to perform their diagnosis, something which is lacking in the rural areas of Africa.

Based on these findings and the survey finding in the region of focus, we concluded that an IDSS that can diagnose malaria and typhoid fever can be based on the practical fact that medicine is evidence based.

The system is able to differentiate different strains of the diseases (i.e. 2 malaria and 2 typhoid strains) in a region, based on its prevalence considering the patient age and travel history. For the prototype, only one strain for each disease was used, as the same treatment applies to both. It was suggested by the physicians who tested the system that it should also incorporate other fibrille diseases like pneumonia and meningitis, so that the application could be used in Wider-African context (i.e. the four Subregional networks). There is an intention to include this in the next incarnation.

The reliability of the system, as compared to both physicians' reference standard, was measured and it was demonstrated that the system could diagnose a disease with a reasonable level of accuracy. The results of the analysis, as shown in figure 6, indicates that typhoid fever bears a closer CNF to the physicians CNF in the majority of cases, and to a lesser degree with cases of malaria. This is as expected because during consultation the certainty of the diagnosis increases as the search criteria is narrowed to signs or symptoms that are very specific to the disease. Typhoid fever has far more discernable signs and symptoms than that of malaria (e.g. diarrhoea is more specific to typhoid fever). Further work is being undertaken to refine the knowledge base for malaria which will overcome these deficiencies.

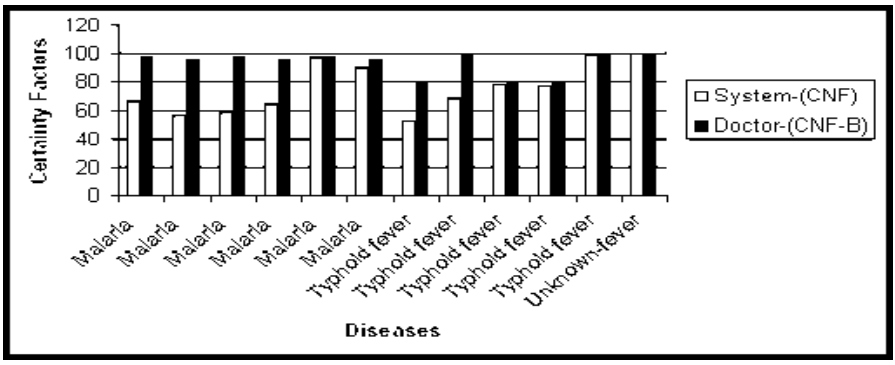


Figure 6 System diagnosis as compared to physicians' reference standard

The present system is a stand-alone application using human judgement and intelligent decision support techniques which provide efficient search strategies to interrogate the knowledge base [19]. It will therefore be suitable, in the future, to mount this on the One Laptop Per Child (OLPC) computer, powered by a wind-up crank or solar panel, making it suitable for the home and healthcare management of malaria and typhoid in rural areas. Thus it overcomes the problems associated withthe lack of basic amenities for the management and storage of available diagnostic tools.

6. Conclusion

Although the system is a relatively straightforward application of medical diagnostics, its method of reasoning provides a more sophisticated data entry approach by eliminating the possibility of entering wrong or conflicting information. This increases its effectiveness when used by novice users, as would be the case in rural areas of the African malaria belt.

The system is simple to learn with little training and this, together with its portability, would make it ideal for people in these rural areas where rural healthcare centres lack the necessary diagnostic tools and medical personnel. Other applications, based upon our demographic findings, would be that the system may be useful for home management of malaria and typhoid fever or for those in a Pharmacy practice where the management of anti-malaria and antibiotic drugs could be based upon the system's diagnosis in cases where a prescription from a physician is not available.

One other important use would also be as a training tool in providing semi-skilled medical assistants with the necessary knowledge and practice to confidently diagnose malaria and typhoid in the early stages.

The benefits of this system can not be overemphasised as the model applied in the Niger-delta region can be applied to the wider Africa malaria subregions as well as other malaria infested regions of the world.

References

1. Webster. D, "Malaria Kills One Child Every 30 Seconds", Journal of Public Health Policy, Vol. 22, No. 1 (2001), pp. 23-33.
2. Unicef, "World Malaria Report 2005- Fact Sheet".
   Source: http://www.unicef.org/media/files/MalariaFactSheet.pdf [online] [Accessed November 02, 2007].
3. ROLLBACKMALARIA.ORG, Source: http://www.rollbackmalaria.org [online] [Accessed January 09, 2008]
4. Dike et al, "Influence of education and knowledge on perceptions and practices to control malaria in Southeast Nigeria", Social science and medicine 63, July 2006, pp. 103-106.
5. Simoes et al, "Performance of health workers after training in integrated management of childhood illness in Gondar, Ethiopia: Bulletin of the World Health Organization", 1997, 75, (Supp l.1), pp. 43-53.

6. Adam Hart-Davies, “What Wateraid Does For Us”, Oasis-the Wateraid journal, Spring/Summer 2006, pp. 6-8. Source: http://www.wateraid.org [Accessed October 15, 2006].
7. Walter, B., In: “$100 Laptop News for the Community ”, OLPC News 2007, One Laptop per Child. Source: http://laptop.org, [online] [Accessed November 02, 2007].
8. Tanyigna et al, “Comparison of blood, bone marrow aspirate, stool and urine cultures in the diagnosis of enteric fever”, Niger J Med. 2001 Jan-Mar; 10(1):21-4.
9. Bojang et al, “A prospective evaluation of a clinical algorithm for diagnosis of malaria in Gambian Children”, Tropical medicine and International health, vol. 5 No. 4 April 2000, pp. 231-236.
10. Shiffman, R. N. and Greenes, R. A., “Improving Clinical Guidelines with Logic and Decision-table techniques: Application to Hepatitis immunization Recommendation”, Medical Decision Making 1994 Vol. 14(3), pp. 245-254.
11. Darlington, K., In: “The Essence of Expert Systems”, 2005, pp. 110-133, Pearson Education Ltd.
12. Winston. P. H., Artificial Intelligence 3rd ed. 1992, pp. 129-132, Addison Wesley.
13. Haque et al, “Early Detection of Typhoid by Polymerase Chain Reaction”, Ann Saudi Med 1999; 19(4): 337-340.
14. Chandramohan, D., et al, “A clinical algorithm for the diagnosis of malaria: results of an evaluation in an area of low endemicity”, Tropical Medicine and International Health, Vol. 6, No. 7, July 2001. pp. 505-510.
15. WHO/USAID, “New perspectives: Malaria Diagnosis”, Report of a joint WHO/USAID informal consultation 25-27, 1999.
Source: http://www.who.int/tdr/cd_publications/pdf/malaria_diagnosis.pdf [online] [Accessed December 01, 2007]
16. Shankar, P., et al, “Decision support systems to identify different species of malarial parasites”, AMIA Annu Symp Proc > v.2003; 2003: 1006.
Source: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1480132 [online] [Accessed March 08, 2008].
17. Anigbogu, S., et al, “Artificial Intelligence-Based Medical Diagnostic Expert System For Malaria And The Related Ailments”, Journal of Computer Science & Its Applications, June 2006, Vol. 12 No. 1,
Source: http://www.ncs.org.ng/pdf/Malaria%201.pdf [online] [Accessed March 08, 2008].
18. Koum, G., et al, “Design of a Two-level Adaptive Multi-Agent System for Malaria Vector driven by an ontology”, BMC Med Inform Decis Mark. 2007, 7:19,
Source: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1925067 [online] [Accessed March 08, 2008].
19. Frawley W., et al, “Knowledge Discovery in Databases: An Overview”, Source: http://www.egeen.ee/u/vilo/edu/200304/DM_seminar_2003_II/Applications/frawley92knowledge.pdf [online] [Accessed March 08, 2008].

Detecting Unusual Changes of Users Consumption

Paola Britos[1], Hernan Grosser[2], Dario Rodríguez[3], and Ramon Garcia-Martinez[4]

Abstract The points being approached in this paper are: the problem of detecting unusual changes of consumption in mobile phone users, the corresponding building of data structures which represent the recent and historic users' behavior bearing in mind the information included in a call, and the complexity of the construction of a function with so many variables where the parameterization is not always known.

1. Introduction

When a mobile call is started, the cells or switches record that it is being made and they produce information referring to this event. These records are commonly called CDR's (Call Detail Records). CDR's contain useful information about the call so that it can be properly charged to whom it may correspond [1]. They can also be used to detect any fraudulent activity considering well-studied fraud indicators. That is, processing an amount of recent CDR's and comparing a function of the different fields such as, IMSI (International Mobile Subscriber Identity, which univocally identifies a user in a mobile phone network), date of call, time of call, duration, type of call (with a specific criteria). If this function retrieves a value that is considered beyond normal limits, an alarm is set off. This alarm must be taken into account by fraud analysts in order to determine if there

[1] Paola Britos
PhD Program, Computer Science School, La Plata University. CAPIS-ITBA. pbritos@itba.edu.ar
[2] Hernan Grosser
Intelligent Systems Lab. School of Engineering. University of Buenos Aires. hgrosser@fi.uba.ar
[3] Dario Rodríguez
Software & Knowledge Engineering Center (CAPIS), ITBA. drodrigu@itba.edu.ar
[4] Ramon Garcia-Martinez
Software & Knowledge Engineering Center (CAPIS), ITBA. rgm@itba.edu.ar

Please use the following format when citing this chapter:

Britos, P., Grosser, H., Rodríguez, D. and Garcia-Martinez, R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 297306.

has been any activity in bad faith or not. To be able to process these CDR's, it is necessary to make previously a process known in telecommunications as mediation, in which the information is read with the format of record in which CDR's come and then it is encoded in a new format of record which is understood by the fraud system.

The existing systems of fraud detection try to consult sequences of CDR's by comparing any field function with fixed criteria known as Triggers. A trigger, when activated, sends an alarm which leads to fraud analysts' investigation. These systems make what is known as a CDR's absolute analysis and they are used to detect the extremes of fraudulent activity. To make a differential analysis, patterns of behavior of the mobile phone are monitored by comparing the most recent activities to the historic use of the phone; a change in the pattern of behavior is a suspicious characteristic of a fraudulent act. [1]

2. Description of the problem

In order to build a system of fraud detection based on a differential analysis it is necessary to bear in mind different problems that arise and must be carefully worked on. These are:

2.1. The problem of building and maintaining "users' profiles"

The majority of fraud indicators are not analyzed by using a unique CDR. In a system of differential fraud detection, information about the history together with samples of the most recent activities is necessary. An initial attempt to solve the problem could be to extract and encode CDR's information and store it in a given format of record. To do this, two types of records are needed; one, which we shall call CUP (Current User Profile) to store the most recent information, and another, to be called UPH (User Profile History) with the historic information [2], [3] and [2]. When a new CDR of a certain user arrives in order to be processed, the oldest arrival of the UPH record should be discarded and the oldest arrival of the CUP should enter the UPH. Therefore, this new, encoded record should enter CUP. This information should be stored in a compact form so it is easy to analyze later on by the system of fraud detection. Considering the amount of information that a CDR contains it is necessary to find a way to "classify" these calls into groups or prototypes where each call must belong to a unique group. This raises several important questions to deal with: (a) What structure must CUP and UPH records have?, (b) How many groups or prototypes must CUP and UPH records have in order to take the necessary information?, (c) How can calls be classified in the different, pre-defined prototypes? and (d) How to encode calls so that they can be "prototyped".

2.2. The problem of detecting changes in behavior

Once the encoded image of the recent and historic consumption of each user is built, it is necessary to find the way to analyze this information so that it detects any anomaly in the consumption and so triggers the corresponding alarm. It is here that the most important question of the whole paper arises and it is: How can the changes in a user's pattern of behavior may be detected? Our problem, then, is focused not only on the detection of abnormal changes in consumption, but also and fundamentally on building the data structures that represent the recent and historic behavior of each user considering the great amount of information that a call takes and the complexity of building a function with so many variables of input, complex and unknown.

3. Description of the suggested solution

The solution that has been developed has taken into account each and every question mentioned before, attempting to solve them in the most effectual and effectively possible way. Below is the presentation of each answer to the questions met in the analysis of the problem. In order to be able start processing the CDR's, a new format of record (mediation process output) must be created containing the following information: IMSI, date of call in YYYYMMDD format, time of the call in HH24MISS format, duration of call in 00000 format and type of call classified as LOC (local call), NAT (national call) and INT (international call). With this information together with the necessary data, it is possible to start solving the following and most important questions by using as input data the output of mediation process.

3.1. User's profiles construction and maintenance Solution

The first point to solve is to determine how to make the CUP and UPH profiles. This means fixing the patterns that will make up each of the profiles. The patterns must have information about the user's consumption, separating LOC consumption (local calls), NAT (national calls) and INT (international calls) respectively. An interesting way to build these patterns is using neural networks so as to determine the space of all users' calls generating a space of patterns which represent the consumption of all users, and then generating a distribution of frequencies by user in which the probability of a user making calls following this pattern is represented [2]. To sum up, when a user's profile is built, the representation of the distribution of frequency in which a certain user makes a certain call is made. This data structure shows the user's pattern of consumption. Among other advantages, neural networks have the capacity to classify the information in certain patterns. Especially, SOM (Self Organizing Map) networks can take this information and build these patterns in a way which is not supervised

by similarity criteria, and without knowing anything a priori about the data [3] and [4]. In our case, all the calls made by all users can be processed so that the networks, depending of the quantity of calls there are of each type, generate the patterns (creating resemblance groups) that represent all of them. To avoid noise in the data, three neural networks are used to generate patterns to represent LOC, NAT, and INT calls respectively. The user's profile is built using all three patterns generated by the three networks. The data used to represent a pattern are the time of the call and its duration. We know that if we represent, in a Cartesian axis, the time of all calls and their corresponding duration, we will obtain a rectangle full of points. The idea is to obtain a graph in which only the most representative points of the whole space will appear; that is the neural network task. Once the patterns that will be used to represent the user's profile are obtained, it is necessary to start filling them with information. The procedure consists of taking the call to be analyzed, encoding it and letting the neural network decide which pattern it resembles. After getting this information, the CUP user profile must be adapted in such a way that the distribution of frequency shows that the user now has a higher chance of making this type of calls. Knowing that a user's profile has K patterns that are made up of L patterns LOC, N patterns NAT and I patterns INT, we can build a profile that is representative of the processed call and then adapt the CUP profile to that call. If the call is LOC, the N patterns NAT and the I patterns INT will have a distribution of frequency equal to 0, and the K patterns LOC will have a distribution of frequency given by the equation [Burge & Shawe-Taylor, 1997a].

$$v_i = \frac{e^{-\|X-Q_j\|}}{\sum_{j=1}^{L} e^{-\|X-Q_j\|}}$$

where:
X: encoded call to be processed
v : probability that X call could be i pattern
Qi: pattern i generated by the neural LOC network.

Notice that: $\sum_{j=1}^{K} v_j = 1$

If the call were NAT, then L must be replaced by N and the distribution of LOC and INT frequencies will be 0; if the call were INT, then L must be replaced by I and the distribution of LOC and NAT frequencies will be 0.

Then, we can define the vector which represents V call, of K dimension, as

Vi = vi, with $1 \le i \le L$

Vi = 0, with $L+1 \le i \le K$, when the call is LOC.

Vi = vi, with $L+1 \le i \le L+N$

Vi = 0 with $1 \le i \le L$ y $L+N \le i \le K$, when the call is NAT.

Vi = vi, with $L+N+1 \le i \le K$

Vi = 0, with $1 \le i \le L+N$, when the call is INT.

Now that we have V vector, we can adapt CUP vector with the information of the processed call:

$CUP_i = \alpha_{LOC} CUP_i - (1- \alpha_{LOC})V_i$, with $1 \le i \le K$, when the call is LOC,

$CUP_i = \alpha_{NAT} CUP_i - (1- \alpha_{NAT})V_i$, with $1 \le i \le K$, when the call is NAT,

$CUP_i = \alpha_{INT} CUP_i - (1- \alpha_{INT})V_i$, with $1 \le i \le K$, when the call is INT, where:

α_{LOC}: adaptability rate applied when call X is incorporated to CUP, if X corresponds to a local call.

α_{NAT}: adaptability rate applied when call X is incorporated to CUP, if X corresponds to a national call.

α_{INT}: adaptability rate applied when call X is incorporated to CUP, if X corresponds to an international call.

Once the CUP profile is adapted, it is compared with the UPH profile and then it is decided whether there has been a significant change in behavior (engine of detection of changes in behavior). After this, the UPH is adapted with the CUP information, only if the number of calls necessary to change the historic patterns has been processed,

$$UPH_i = \beta UPH_i + (1-\beta)CUP_i$$

With $1 \leq i \leq K$, where β: adaptability rate applied when CUP is incorporated to UPH.

3.2. Solution to the detection of changes in behavior

In order to settle whether there have been changes in the pattern of behavior or not, it is necessary to compare, somehow, the CUP and UPH profiles and decide if the difference between them is big enough so as to set an alarm off. Because both the CUP and the UPH are vectors that represent frequency distributions, a vectorial distance can be used to compare how different they are. For this, the Hellinger distance (H) can be used; it indicates the difference between two distributions of frequency [1]. This distance will always be somewhere between zero and two, where zero is for equal distributions and two represents orthogonally. The value of H will establish how different must CUP and UPH frequency distributions be, in order to set an alarm going. By changing this value, there will be more or fewer alarms set off.

3.3. Limitations of the solution

This solution is focused, as we described, on the analysis of the user's differential consumption. One case that may not be detected would be that in which the user always makes a lot of calls of the same type with a high consumption, as his pattern of behavior would never change. That is why there should always be a combination of several solutions in order to have a system of fraud detection that can detect different types of fraud. In this case, the absolute analysis would be a good solution. The other limitation centers in that the patterns are static, so that if the way in which the company users consume changes completely, it will be necessary to train again the neural networks to establish new patterns that represent the total space of calls and to re- build the CUP and UPH profiles as from the new distributions.

4. Experimentation

4.1. Methodology used

The experiments were divided in two parts: the first was focused on the training of the neural network and the generation of patterns to build the user's profiles later on; the second was aimed at the analysis of the calls made by high-consumption users and the corresponding analysis and detection of alarms. The second part of the test was divided again into two different experiences: 1) updating of UPH profile with each call (f= 1 call) and low Hellinger threshold (H) for the setting off of alarms of change of behavior; 2) updating of UPH profile once a day (f= 1 day) and high Hellinger threshold (H).

4.2. Experiments on the generation of patterns

Three SOM were built for the generation of patterns for LOC, NAT and INT calls respectively. Each of the networks was trained with an amount of calls that was representative of the consumption that company users made during a couple of days at all times. The calls were introduced to the network in a disorderly manner so that the patterns that were generated were not representative only of the time and duration of the last calls. The result of this experience defined the patterns to build the users' profiles. The patterns are made up of the time of the call and its duration in minutes, which managed to build a discrete space composed of all the types of called made by any user in a fixed quantity representative of that space.

4.3. Experiments on the construction of profiles and detection of behaviors

Once the patterns that define the space of all calls are obtained, tests have been carried out on the construction of user's profiles through the development of a distribution of frequencies of each of the patterns for each profile (CUP and UPH) and the corresponding detection of alarms. The process was based on the introduction to the system of calls made within a period of three months by users reported as "high-consumption user". With each call the CUP user profile was updated, it was then compared with the UPH profile, thus, obtaining the Hellinger distance between them. If it surpassed the fixed threshold, an alarm was set off. Depending on the parameter of updating frequency of UPH profile (f), the UPH was updated with the corresponding contribution of the CUP. At the moment of inputting a user's first call, all CUP and UPH patterns were initialized with the same distribution of frequency, assuming a priori that the user had the same

tendency to make any type of call, without any information. Moreover, this experience was carried out twice; the first one updating the UPH with each call, therefore, with a low Hellinger threshold (H) for the detection of alarms. This was because the difference that may arise between the CUP and UPH profiles was too small if updating the historic profile with each call, due to the fact that the historic profile tended to be the same as the current profile. The second experience was made by updating the UPH once a day and a high Hellinger (H) threshold to detect important differences that can be considered as changes in behavior.

5. Results

5.1. Generation of patterns

In this section results are presented after the training of the three SOM (See Fig. 1 to 3). The results show each of the patterns that the networks fixed as most representative of the space of all the users' calls. Three graphs are represented (one for each network) to show the patterns that were generated. On axis X, the time of the call is shown and on axis Y, the duration expressed in minutes is illustrated. Each of the points represented corresponds to a pattern being chosen by the network as representative of the sample. In the local neural network graph, 144 patterns are shown, in the NAT network, 64 and in the INT network, 36.

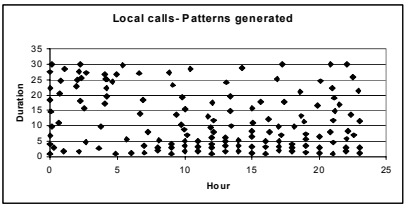


Fig. 1. Patterns generated after the training corresponding to local calls

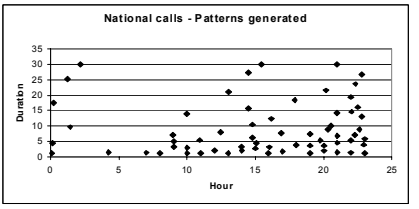


Fig. 2. Patterns generated after the training corresponding to national calls

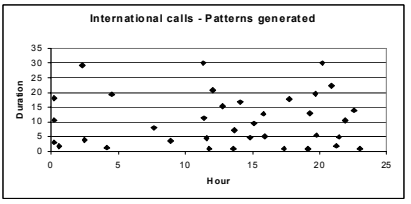


Fig. 3. Patterns generated after the training corresponding to international calls

The graph (Fig. 1) shows the 144 patterns generated after the training of the neural network corresponding to local calls. At simple sight, it is easy to notice that there is a greater concentration of patterns in the time range between 8h and 20h and duration of about 0–5 minutes. This denotes that most of the local calls made by this company's customers occur at these hours with the average durations indicated. The graph (Fig. 2) shows the 64 patterns generated after the training of the national calls neural network. Here, also, a concentration of patterns can be seen, but this time more towards the time range of 15h to 22h with durations that vary between 0 and 7 minutes. It also shows that there are practically no patterns generated for dawn, which may lead to conclude that most users of the company being analyzed do not make any NAT calls during early hours. The graph (Fig. 3) shows the 36 patterns after the training of the international calls neural network. Here the distribution is a little more aleatory, but the duration of the calls "chosen" as patterns tends to have a longer duration (between 7 and 10 minutes).

5.2. Profiles construction and changes detection in behavior

In this section, the results presented were obtained after the construction (from the company records) of the profiles and the detection of the corresponding alarms for each of the two experiences made. The graphs show the CUP and UPH profiles at the moment an alarm was set off. On axis X, the 244 patterns (144 LOC, 64 NAT and 36 INT) are shown and on axis Y the distribution of frequencies of each of the patterns for the user being analyzed at the moment the alarm was set off.

5.2.1. Experience 1 (Updating UPH with each call, high sensitivity with low Hellinger Threshold)

The graph (Fig. 4) shows a user's CUP at the moment an alarm was set off. It can be observed that the distribution of frequencies indicates a major tendency to make NAT calls (patterns 145 to 208). The graph (Fig. 5) shows the same user's UPH at the moment the alarm was set off. It can also be observed that the distribution of frequencies indicates a major tendency to make local calls (patterns 1 to 144). Hence, the difference between both distributions of frequencies defined by Hellinger distance (H) equals 0,30081. By analyzing the detail of this user's calls from dates previous to the triggering of the alarm to the day it was set off, there is evidence that the alarm responded to the user's making his first NAT call since his calls were processed.

This means, his historic pattern of behavior did not make it evident that this user would make such a call. However, when these calls were made, the system detected the change and generated the corresponding alarm. These results also show that, having made the experience with such high sensitivity, one single different call can indicate a change in behavior that leads to an alarm. The total number of alarms that were set off after analyzing the 60 users was 88, out of which 33 correspond to different cases.

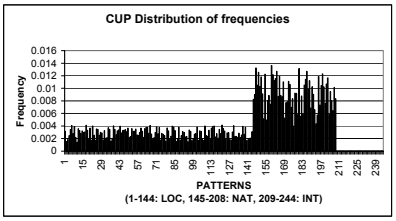


Fig. 4. User's CUP at the moment an alarm was set off

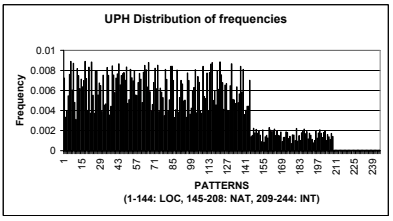


Fig. 5. User's UPH at the moment an alarm was set off

This is due to the fact that once an alarm for a user is set off, the following calls keep on setting off alarms till the UPH definitely adapts to the change in behavior. Most of the calls follow the pattern of the case in the graph in which a call that is different from the normal pattern of behavior is enough for the system to define the user as suspicious.

5.2.2. Experience 2 (Updating UPH once a day, moderate sensitivity with Hellinger threshold)

The graph (Fig. 6) shows a user's CUP at the moment an alarm was set off. It can be observed that the distribution of frequencies indicates a tendency to make local calls (patterns 1 to 144) and International calls (patterns 209 to 244). The graph (Fig. 7) shows the same user's UPH at the moment the alarm was set off.

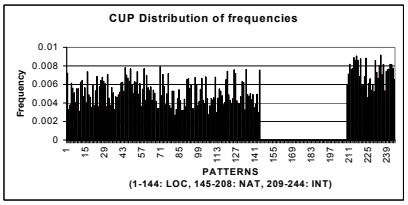


Fig. 6. User's CUP at the moment an alarm was set off

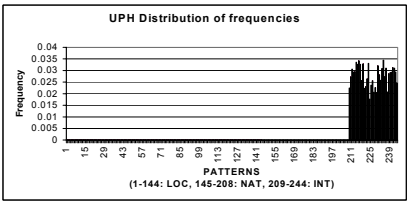


Fig. 7. User's UPH at the moment an alarm was set off

It can also be observed that the distribution of frequencies indicates a major tendency to make INT calls only (patterns 209 to 244). Therefore, the difference between both distributions of frequencies defined by Hellinger distance (H) equals 0,82815. By analyzing the detail of this user's calls from dates previous to the triggering of the alarm to the day it was set off, there is evidence that the alarm responded to the user's making only international calls till the moment that he started making local calls. When the number of local calls modified the CUP in the way illustrated by the graph, the alarm was triggered. This is a particular case as; surely, this alarm is not an indicator of fraud if the user pays his invoice for international calls. But it is an indicator of a sensitive change of behavior in the pattern of consumption, and that is exactly what this system searches. The total

number of alarms that were set off after analyzing the 60 users was 64, out of which 14 correspond to different cases. This is due to the fact that once an alarm for a user is set off, the following calls keep on setting off alarms till the UPH definitely adapts to the change in behavior. This phenomenon is emphasized here because it is only after calls of the next day are processed that the UPH is updated. The majority of the calls follow the pattern of the case in the graph in which there must be several calls outside the pattern of behavior for the system to find the user suspicious. This is much more satisfactory than what was obtained in experience 1 in which the high sensitivity showed users as suspicious simply for having made one single different call.

6. Conclusions

The results that were obtained were satisfactory in the sense that they were able to establish changes in the behavior of the users analyzed. Though the change in behavior does not necessarily imply fraudulent activity, it manages to restrict fraud analysts' investigation to this users' group. By using then other types of techniques [5], it is possible to obtain, with a high degree of certainty, a list of users who are using their mobile phone in an "unloyal" way. Besides, the experiences have helped to find users who have effectively changed their behavior, but in an inverse way, i.e. , they were users with high INT consumption and then they started making local calls. Commercially speaking, it could be an interesting tip to evaluate this type of consumers, since, for a certain reason they decided not to use their mobile phones to make international calls any more and it could help draw conclusions and create new rate plans based on these situations. It is also proven, with the experiences carried out, that the differential analysis provides with much more information than the absolute analysis, which can only detect peaks of consumption and cannot describe the user in question. As a final conclusion, neural networks can be said to be excellent tools for the classification of calls and the construction of users' profiles as they represent their behavior in a faithful and efficient manner.

References

1. ASPeCT, *Definition of Fraud Detection Concepts*, Deliverable D06. 47 pages. (1996).
2. Burge P, Shawe-Taylor J. *Fraud Detection and Management in Mobile Telecommunications networks*, Department of Computer Science Royal Holloway, University of London. Vodafone, England. Siemens A. G. (1997).
3. Kohonen, T. *Self-Organizing Maps*. Springer Series in Information Sciences. (2000).
4. Hollmen J. *Process Modeling using the Self-Organizing Map*, Helsinki University of Technology Department of Computer Science. (1996).
5. ASPeCT, *Fraud Management tools: First Prototype*, Deliverable D08. 31 pages. (1997).

TECHNIQUES

Optimal Subset Selection for Classification through SAT Encodings

Fabrizio Angiulli and Stefano Basta

Abstract In this work we propose a method for computing a minimum size training set consistent subset for the Nearest Neighbor rule (also said CNN problem) via SAT encodings. We introduce the SAT–CNN algorithm, which exploits a suitable encoding of the CNN problem in a sequence of SAT problems in order to exactly solve it, provided that enough computational resources are available. Comparison of SAT–CNN with well-known greedy methods shows that SAT–CNN is able to return a better solution. The proposed approach can be extended to several hard subset selection classification problems.

1 Introduction

Most useful classification tasks can be formulated as subset selection problems [6, 17, 12, 26, 28]. Subsets to be singled out have to posses certain properties guaranteeing that they represent a model of the whole training set, according to the specific classification rule. Often the number of potential models is exponential in the training set size and, among all the training set subsets, the *optimal model* is that composed of the minimum number of objects. Indeed, a small model improves both response time and (according to the Occam's razor) generalization.

For example, a *sample compression scheme* [12] is defined by a fixed rule $\rho : T \mapsto \rho(T)$ for constructing a classifier from a given set of data T. Given a training set T, it is compressed by finding the smallest subset $S \subseteq T$ for which the classifier $\rho(S)$ correctly classifies the whole set T. It is known that the size of a sample compression scheme can be used to bound generalization.

Fabrizio Angiulli
DEIS, Università della Calabria, Via P. Bucci 41C, 87036 Rende (CS), Italy, e-mail: f.angiulli@deis.unical.it

Stefano Basta
ICAR-CNR, Via P. Bucci 41C, 87036 Rende (CS), Italy, e-mail: basta@icar.cnr.it

Please use the following format when citing this chapter:

Angiulli, F. and Basta, S., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 309–318.

Unfortunately, minimum cardinality subset selection problems often turn out to be intractable (e.g., [27]). Consequently, authors provide greedy heuristics (e.g., [18]) or attempt to search for near optimal solutions using non exhaustive search methods (e.g., [7]) or semi-naive enumeration methods (e.g., [23]) .

Nearest neighbor condensation. The Nearest Neighbor (NN rule for short) decision rule [6] is a widely employed classification rule. The NN rule assigns to an unclassified sample point the classification of the nearest of a set of previously classified points. For this decision rule, no explicit knowledge of the underlying distributions of the data is needed. A strong point of the NN rule is that, for all distributions, its probability of error is bounded above by twice the Bayes probability of error [6, 24, 10].

Naive implementation of the NN rule requires storage of all the previously classified data, and then comparison of each sample point to be classified to each stored point. In order to reduce both space and time requirements, several techniques to reduce the size of the stored data for the NN rule have been proposed (see [28] and [25] for a survey) referred to as training set condensation algorithms. In particular, among these techniques, *training set consistent* ones, aim at selecting a subset of the training set that classifies the remaining data correctly through the NN rule.

According to the discussion above, using a training set consistent subset, instead of the entire training set, to implement the NN rule, has the additional advantage that it may guarantee better classification accuracy. Indeed, [19] showed that the VC dimension of an NN classifier is given by the number of reference points in the training set. Moreover, computing a minimum cardinality training set consistent subset for the NN rule has been shown to be intractable [27].

A number of greedy training set condensation algorithms have been proposed that extract a consistent subset of the overall training set, namely CNN, RNN, MCNN, NNSRM, FCNN, and others [18, 15, 19, 9, 1, 3]. Approximate optimization methods, such as tabu search, gradient descent, evolutionary learning, and others, have been used to compute subsets close to the minimum cardinality one: [20] provides a comparison of a number of these techniques. However, none of these algorithms guarantees that the solution returned is of minimum size.

SAT Encodings. The SAT Problem [5] consists in deciding whether for a given Boolean formula there exists a truth value assignment to its variables that makes the formula true. SAT is the archetypical problem for the NP complexity class [14] and, therefore, many problems of practical interest in, among other examples, artificial intelligence, operations research, and electronic design engineering, can be SAT encoded, that is translated in suitable instances of SAT.

SAT solver technology is emerging, as witnessed by the annual conference devoted to this theme (the International Conferences on Theory and Applications of Satisfiability Testing are the primary annual meetings for researchers studying the SAT problem[1]), by several SAT solver implementations (e.g., [11, 21, 22]), and by

[1] See `http://www.satisfiability.org/`.

the annual competition (the international SAT Competitions identify new challenging benchmarks, promote new solvers for the SAT problem as well as compare them with state-of-the-art solvers[2]).

Proposed approach. In this work we investigate the possibility of computing a minimum size training set consistent subset for the NN rule (the CNN problem) via SAT encoding. The CNN problem is NP-hard [27] and belongs to the complexity class $\mathrm{FP}^{\mathrm{NP}[O(\log n)]}$, that is, loosely speaking, the class of the problems that can be solved in polynomial time by invoking at most a logarithmic number of times a procedure able to solve a problem in NP and which is assumed to reply instantaneously.

Basing on this property, we introduce the SAT–CNN algorithm, which exploits a suitable encoding of the CNN problem in a sequence of SAT problems in order to exactly solve it, provided that enough computational resources are available. The proposed approach can be extended to several intractable subset selection classification problems, such as SNN [23], k-NN [13], k-center [17], CNNDD [2], and others.

The rest of the work is organized as follows. In Section 2 some preliminary definitions are provided. Section 3 describes the SAT–CNN algorithm. Section 4 reports some experimental results. Finally, Section 5 depicts conclusions and future works.

2 Preliminary Definitions

In the following by T a labelled training set from a space $\mathscr{S}$ with distance d is denoted.

Let x be an element of T. By $nn(x,T)$ the nearest neighbor of x in T according to the distance d is denoted. By $\ell(x)$ the label associated to x is denoted.

Given a labelled data set T and an element y of $\mathscr{S}$, the *nearest neighbor rule* $\mathrm{NN}(y,T)$ assigns to y the label of the nearest neighbor of y in T, i.e. $\mathrm{NN}(y,T) = \ell(nn(y,T))$ [6].

A subset S of T is said to be a *training set consistent subset of* T if, for each $x \in T$, $\ell(x) = \mathrm{NN}(x,S)$ [18].

Given a training set T, the Minimum Training Set Consistent Subset Problem (or CNN problem) $\langle T \rangle$ is as follows: return a training set consistent subset S^* of T such that, for any other training set consistent subset S of T, $|S^*| \leq |S|$.

Given a training set T and a positive integer number k, the Training Set Consistent Subset Problem (or k-CNN problem) $\langle T,k \rangle$ is as follows: return a training set consistent subset S of T such that $|S| \leq k$, if at least one exists, and the empty set, otherwise.

Given a training set T and a positive integer number k, the decision version $\langle T,k \rangle_D$ of the problem $\langle T,k \rangle$ is as follows: return "no" if the answer of $\langle T,k \rangle$ is the empty set, and "yes" otherwise.

[2] See `http://www.satcompetition.org/`.

3 The SAT–CNN Algorithm

The algorithm SAT–CNN computes a minimum size training-set consistent subset of the input training set T. It accomplishes its task by encoding the problem of computing a training set consistent subset in a sequence of suitable instances of the SAT problem. Without loss of generality the Boolean formula is in conjunctive normal form (CNF), that is it is the conjunction of one or more clauses. A clause is the disjunction of one or more literals. A truth value assignment σ to the set of variables $X = \{x_1, \ldots, x_n\}$ is a function $\sigma : X \mapsto \{true, false\}$.

3.1 SAT Encoding

Given a labelled training set $T = \{o_1, \ldots, o_n\}$ and a positive integer number k, in the following $\Psi^k(T)$ denotes the SAT encoding of the decision problem $\langle T, k \rangle_D$.

More precisely, $\Psi^k(T)$ is a Boolean formula in conjunctive normal form[3] defined on the set of variables $x_1, x_2, \ldots, x_n$ and on some others auxiliary variables that will be introduced next. In particular, each variable x_i is associated with the object o_i of the training set T ($1 \leq i \leq n$). Indeed, if a truth value assignment for the variables in the formula $\Psi^k(T)$ makes the formula true, then the variables $x_1, \ldots, x_n$ encode a training set consistent subset S of T. In particular, the variable x_i being true (false, resp.) means that the corresponding object o_i belongs (does not belong, resp.) to S.

The formula $\Psi^k(T)$ consists of two sets of clauses, namely $\Psi_{cons}(T)$, also called *constraint clauses*, and $\Psi^k_{size}(T)$, also called *cardinality clauses*.

The clauses in the set $\Psi_{cons}(T)$ serve the purpose of guaranteeing that the subset S encoded by the truth assignment for the variables x_i is indeed a training set consistent subset of T. These clauses do not depend on the positive integer k.

The clauses in the set $\Psi^k_{size}(T)$ serve the purpose of guaranteeing that the size of subset S encoded by the truth assignment for the variables x_i does not exceed the value k.

Next we describe the structure of the two above introduced set of clauses. We assume that the training set T consists of $m \geq 2$ class labels $l = 1, 2, \ldots, m$, and that n_l represents the number of objects belonging the the class l (clearly, $n_1 + n_2 + \ldots + n_m = n$).

3.1.1 Constraint clauses

Before describing the constraint clauses, the following preliminary definition is needed.

[3] In the following we will use the terms *boolean formula in conjunctive normal form* and *set of clauses* interchangeably .

Given a labelled training set T and two objects o_i and o_j of T, having different class labels, by $c(o_i, o_j)$ we denote the set of objects of T which have the same class label of o_i and whose distance from o_i is not greater than the distance from o_i to o_j, that is

$$c(o_i, o_j) = \{o \in T \mid \ell(o) = \ell(o_i) \wedge \mathrm{d}(o, o_i) \leq \mathrm{d}(o_j, o_i)\}.$$

In order to guarantee that the truth value assignment for the set of variables $x_1, x_2, \ldots, x_n$ encodes a training set consistent subset S of T, it must be the avoided that there exist two objects o_i and o_j having different class labels, such that o_j belongs to S and o_j misclassifies o_i. The object o_i is not misclassified by the object o_j if there exists an object o_h in the set S, having the same class label of o_i and whose distance from o_i is less than the distance from o_i to o_j. As a whole the following property must be verified:

$$\begin{aligned}(\forall o_j)(\forall o_i)(\ell(o_j) \neq \ell(o_i) \wedge o_j \in S) \rightarrow \\ (\exists o_h \in S)(\ell(o_h) = \ell(o_i) \wedge \mathrm{d}(o_i, o_h) \leq \mathrm{d}(o_i, o_j)),\end{aligned}$$

that can be encoded through the following set of clauses

$$r_{i,j}^{(1)} \equiv x_j \rightarrow \left(\bigvee_{o_h \in c(o_i, o_j)} x_h \right) \equiv \neg x_j \vee \left(\bigvee_{o_h \in c(o_i, o_j)} x_h \right), \tag{1}$$

where i and j are such that $1 \leq i \leq n$, $1 \leq j \leq n$, and $\ell(o_i) \neq \ell(o_j)$. The number of clauses (1) is $\sum_{l=1}^{m} n_l(n - n_l) = \mathscr{O}(n^2)$ and each of them is composed of at most $1 + \max_{l=1}^{m} n_l = \mathscr{O}(n)$ literals. Thus, overall, clauses (1) are composed of at most $\mathscr{O}(n^3)$ literals.

Note that the truth value assignment which assigns false to every variable x_i satisfies clauses (1). Nonetheless, the empty set is not a valid training set consistent subset. Thus, the following set of clauses is needed in order to enforce nonemptyness of the solution set S:

$$r_l^{(2)} \equiv \bigvee_{o_i : \ell(o_i) = l} x_i, \tag{2}$$

where $l \in \{1, 2, \ldots, m\}$. The number of clauses (2) is m and, as a whole, they are composed of exactly n literals. In particular, clauses (2) require that for each class label at least one object of that class belongs to S. Clauses (1) and (2) form the set $\Psi_{cons}(T)$, which guarantees that S is a training set consistent subset of T.

Before concluding the description of the constraint clauses, it is important to point out that the truth value assignment which assigns true to every variable x_i, trivially satisfies all the clauses in $\Psi_{cons}(T)$. As a matter of fact, T is always a training set consistent subset of itself. Cardinality rules, described in the following, will take care of upper bounding the size of the subset S.

3.1.2 Cardinality clauses

The formula $\Psi^k_{size}(T)$ is defined on the n variables $x_1, x_2, \ldots, x_n$ and also on the nk auxiliary variables $e_{i,j}$, $i = 1, \ldots, n$, $j = 1, \ldots, k$. In particular, the variable $e_{i,j}$ being true (false, resp.) encodes the fact that the object o_i of T is (is not, resp.) the j-th element of the set S.

The clauses composing the set $\Psi^k_{size}(T)$ are detailed next.

First of all, it must be guaranteed that if o_i is the j-th element of S then x_i belongs to S (nk clauses of size 2):

$$r^{(3)}_{i,j} \equiv e_{i,j} \rightarrow x_i \equiv \neg e_{i,j} \vee x_i, \tag{3}$$

where $i = 1, \ldots, n$ and $j = 1, \ldots, k$.

Furthermore, if o_i belongs to S then it must exists a value $j_i \in \{1, 2, \ldots, k\}$ such that x_i is the j_i-th element of S (n clauses of size $k+1$):

$$r^{(4)}_i \equiv x_i \rightarrow \left(\bigvee_{j=1}^{k} e_{i,j} \right) \equiv \neg x_i \vee \left(\bigvee_{j=1}^{k} e_{i,j} \right), \tag{4}$$

where $i = 1, \ldots, n$.

Given Boolean variables $y_1, \ldots, y_n$, the *at-most-one constraint*

$$\textbf{at-most-one}(y_1, \ldots, y_n)$$

is a set of clauses which is satisfied if and only if at most one of the variables $y_1, \ldots, y_n$ is true.

The two following sets of clauses are needed to complete the cardinality clauses. The object o_i may occur at most one time in the subset S, that is, for each $i = 1, \ldots, n$,

$$r^{(5)}_i \equiv \textbf{at-most-one}\{e_{i,1}, \ldots, e_{i,k}\}, \tag{5}$$

and the j-th element of S may be at most one of the elements of T, that is, for each $j = 1, \ldots, k$,

$$r^{(6)}_j \equiv \textbf{at-most-one}\{e_{1,j}, \ldots, e_{n,j}\}. \tag{6}$$

Note that the formula $\Psi^k_{size}(T)$ enforces the set S to have at most k elements, hence S could be composed of less than k elements.

The at-most-one constraint can be formulated in different ways. Here we make use of the formulation known as *ladder encoding* [16, 4]. The ladder encoding of the at-most-one constraint $\textbf{at-most-one}(y_1, \ldots, y_n)$ is the Boolean formula, defined on the variables $y_1, \ldots, y_n$ and also on n novel variables $z_1, \ldots, z_n$, composed of the following $\mathcal{O}(n)$ clauses: the *ladder validity* clauses, for $i = 2, \ldots, n$,

$$c'_i \equiv z_i \rightarrow z_{i-1} \equiv \neg z_i \vee z_{i-1},$$

and the *channeling* clauses, for $i = 1, \ldots, n$,

Algorithm SAT–CNN
Input: a training set T and a timeout τ
Output: a training set consistent subset S_{opt} of T

1. Compute the *constraint clauses* $\Psi_{cons}(T)$
2. Optionally use a greedy method to find a seed cardinality training-set consistent subset S_{seed}, having size k_{seed}; otherwise set S_{seed} to T and k_{seed} to the size n of T
3. Set $k_{max} = k_{seed}$, $k_{min} = m$, $k_{opt} = k_{seed}$, $S_{opt} = S_{seed}$, and *approx* to false
4. If $k_{min} > k_{max}$ then goto 12
5. Set $k_{curr} = (k_{min} + k_{max})/2$
6. Compute the *cardinality clauses* $\Psi_{size}^{k_{curr}}(T)$
7. Solve the SAT problem $\Psi^{k_{curr}}(T) = \Psi_{cons}(T) \cup \Psi_{size}^{k_{curr}}(T)$
8. If the answer to $\Psi^{k_{curr}}(T)$ is "yes", then determine the size k_{sol} of the assignment $\sigma_{k_{curr}}$ found, that is the number of variables x_i which evaluate to true in $\sigma_{k_{curr}}$, and set $k_{max} = k_{sol}$, $S_{opt} = \{o_i \mid \sigma_{k_{curr}}(x_i) = true\}$, and $k_{opt} = k_{sol}$
9. If the answer to $\Psi^{k_{curr}}(T)$ is "no", then set $k_{min} = k_{curr} + 1$
10. If the answer to $\Psi^{k_{curr}}(T)$ is "unknown", then set $k_{min} = k_{curr} + 1$ and *approx* to true
11. Goto 4
12. Return the training set consistent subset S_{opt} and its size k_{opt}. If *approx* is set to true than the solution is approximate

Fig. 1 The Algorithm SAT–CNN.

$$c_i'' \equiv y_i \leftrightarrow (z_i \wedge \neg z_{i+1}) \equiv (y_i \vee \neg z_i \vee z_{i+1}) \wedge (\neg y_i \vee z_i) \wedge (\neg y_i \vee \neg z_{i+1}).$$

Intuitively, clauses c_i' impose that each truth value assignment for the variables $z_1, \ldots, z_n$ is of the form

$$(z_1, \ldots, z_t, z_{t+1}, \ldots, z_n) = (true, \ldots, true, false, \ldots, false),$$

where the number t of variables which evaluates to true can be zero, one, or more than one, while clauses c_i'' guarantee that y_t is true (if t is zero then no variable y_i is true).

3.2 SAT–CNN Algorithm

The algorithm SAT–CNN is a binary search based method enhanced with a greedy initialization step and exploiting the size of the current solution in order to accelerate convergence.

The algorithm is reported in Figure 1. Step 1 computes the constraint clauses $\Psi_{cons}(T)$. During the main cycle (steps 4-11) the minimum cardinality subset is searched for by adaptively adjusting the value of cardinality k_{curr} and then solving the SAT problem $P_{k_{curr}} = \Psi_{cons}(T) \cup \Psi_{size}^{k_{curr}}(T)$.

Other than the training set T, the algorithm receives in input a timeout τ, denoting the maximum amount of time allowed to the SAT solver to solve the current instance $\Psi^{k_{curr}}(T)$. If the solver does not return an answer to $\Psi^{k_{curr}}(T)$ within time τ, then it is stopped and its answer is assumed to be "unknown" (see step 10). Note that the solver may return the answer "unknown" also because either the memory is over or it is not able to answer to the given instance (the latter situation may occur only when the solver is not complete).

4 Experimental Results

We interfaced SAT-CNN with the RSat 2.0 SAT solver [22]. RSat is a DPLL-based [8] complete SAT solver that employs many modern techniques such as those used in MiniSat [11] and Chaff [21]. It won gold medals from the SAT'07 competition in the industrial category.

We compared the cardinality of the solution computed by the SAT–CNN algorithm with the cardinality of the solutions returned by well-known greedy algorithms, namely CNN, MCNN, NNSRM, and FCNN [18, 19, 9, 1, 3].

In the experiments, S_{seed} was always set to the whole training set (see step 2 in Figure 1), while the timeout τ was set to 500 seconds. We employed a Core 2 Duo based machine having 2GB of main memory.

The next table reports the data set employed in the experiments (data sets are from the UCI Machine Learning Repository[4]), together with the size of the solution computed by SAT–CNN compared with the best size returned by the greedy algorithms.

Data Set	*Size*	*Dims*	*Classes*	SAT–CNN	*Greedy*	*Ratio*
Bupa	345	6	10	145	168	86%
Colon Tumor	62	2,000	2	13	17	76%
Echocardiogram	61	11	2	2	5	40%
Iris	150	4	3	10	13	77%
Ionosphere	351	34	2	45	55	82%
Pima	768	8	2	300	316	95%
SPECT Heart	349	44	2	75	93	81%
Vehicle	846	18	4	348	382	91%
Wine	178	13	3	51	62	82%

The last column shows the ratio between the size of the SAT–CNN solution and the size of the best greedy solution. The SAT-CNN algorithm improved over greedy methods in all cases. Moreover, it reported that the solution is exact on the *Iris* and *Echocardiogram* data sets.

[4] See `http://mlearn.ics.uci.edu/MLRepository.html`.

Data Set	*Time 1*	*Time 2*	*Clauses*	*Max. Clauses*	*Max. Vars*
Bupa	3,021	64	3,811,018	593,440	178,882
Colon Tumor	1,140	7	49,984	19,304	5,920
Echocardiogram	6	3	20,432	18,210	5,642
Iris	665	31	220,825	116,849	34,124
Ionosphere	2,529	43	1,683,093	610,929	185,152
Pima	39,169	717	19,827,074	2,925,278	886,655
SPECT Heart	2,151	38	2,149,965	596,190	183,050
Vehicle	3,661	623	29,753,120	3,768,274	1,078,225
Wine	2,564	10	709,579	164,147	47,970

Finally, we report in the table above some statistics concerning SAT–CNN, that are the total execution time (column *Time 1*, in seconds), the rewriting time (column *Time 2*, in seconds), the total number of clauses evaluated (column *Clauses*), and the maximum number of clauses (column *Max. Clauses*) and variables (column *Max. Vars*) included in a single SAT instance.

5 Conclusions and Future Work

This work introduces the SAT–CNN algorithm, which exploits a suitable encoding of the CNN problem in a sequence of SAT problems in order to exactly solve it.

As future work we plan to extend experiments in order to study how the size of the solution varies with the timeout, to take into account other training sets, to investigate testing accuracy, and to compare with approximate optimization approaches. We also plan to run our method with other state of the art SAT solvers, and to provide encodings for other families of solvers, such as *pseudo-boolean solvers* and *stable models engines*. We will also investigate alternative rewritings for the cardinality clauses and methods to reduce the number of constraint clauses. Finally, we will extend the method here presented to other classification tasks that can be formalized as hard subset selection problems, as SNN [23], k-NN [13], k-center [17], CNNDD [2], and others.

References

1. Angiulli, F. (2005). Fast condensed nearest neighbor rule. In *22nd International Conference on Machine Learning (ICML)*, Bonn, Germany.
2. Angiulli, F. (2007). Condensed nearest neighbor data domain description. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(10):1746–1758.
3. Angiulli, F. (2007). Fast nearest neighbor condensation for large data sets classification. *IEEE Trans. Knowl. Data Eng.*, 19(11):1450–1464.
4. Manyà, F., & Ansótegui, C (2004). Mapping problems with finite-domain variables into problems with boolean variables. In *Proc. of the Seventh Int. Conf. on Theory and Applications of Satisifiability Testing* (SAT), pages 111–119, Vancouver, BC, Canada.
5. Cook, S.A. (1971). The complexity of theorem-proving procedures. In *3rd ACM Symposium on Theory of Computing*, pages 151–158, Ohio, United States.
6. Hart P.E., & Cover, T.M. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

7. Dasarathy, B. (1994). Minimal consistent subset (mcs) identification for optimal nearest neighbor decision systems design. *IEEE Transactions on Systems, Man, and Cybernetics*, 24(3):511–517.
8. Logemann, G., Loveland, D., & Davis, M. (1962). A machine program for theorem-proving. *Communications of the ACM*, 5(7):394–397.
9. Murty, M.N., & Devi, F.S. (2002). An incremental prototype set building technique. *Pattern Recognition*, 35(2):505–513.
10. Devroye, L. (1981). On the inequality of cover and hart in nearest neighbor discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:75–78.
11. Sörensson, N., & Eén, N. (2005). Minisat a sat solver with conflict-clause minimization. In *International Conference on Theory and Applications of Satisfiability Testing*.
12. Warmuth, M., & Floyd, S. (1995). Sample compression, learnability, and the vapnik-chervonenkis dimension. *Machine Learning*, 21(3):269–304.
13. Hostetler, L.D., & Fukunaga, K. (1975). *k*-nearest-neighbor bayes-risk estimation. *IEEE Trans. on Information Theory*, 21:285–293.
14. Johnson, D.S., & Garey, M.R. (1979). *Computers and Intractability. A Guide to the Theory of NP-completeness*. Freeman and Comp., NY, USA.
15. Gates, W. (1972). The reduced nearest neighbor rule. *IEEE Transactions on Information Theory*, 18(3):431–433.
16. Prosser, P., & Gent, I.P. (2002). In *Proc. of the Fifth Int. Conf. on Theory and Applications of Satisifiability Testing* (SAT), Cincinnati, Ohio, USA.
17. Gonzalez, T. (1985). Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306.
18. Hart, P.E. (1968). The condensed nearest neighbor rule. *IEEE Transactions on Information Theory*, 14(3):515–516.
19. Krim, H., & Karaçali, B. (2003). Fast minimization of structural risk by nearest neighbor rule. *IEEE Transactions on Neural Networks*, 14(1):127–134.
20. Nakagawa, M., & Liu, C.L. (2001). Evaluation of prototype learning algorithms for nearest-neighbor classifier in application to handwritten character recognition. *Pattern Recognition*, 34(3):601–615.
21. Madigan, C., Zhao, Y., Zhang, L., Malik, S., & Moskewicz, M. (2001). Engineering an efficient sat solver. In *39th Design Automation Conference (DAC)*.
22. Darwiche, A., & Pipatsrisawat, K. (2007). Rsat 2.0: Sat solver description. Technical Report D–153, Automated Reasoning Group, Computer Science Department, UCLA.
23. Woodruff, H.B., Lowry, S.R., Isenhour, T.L., & Ritter, G.L. (1975). An algorithm for a selective nearest neighbor decision rule. *IEEE Transactions on Information Theory*, 21:665–669.
24. Stone, C. (1977). Consistent nonparametric regression. *Annals of Statistics*, 8:1348–1360.
25. Toussaint, G. (2002). Proximity graphs for nearest neighbor decision rules: Recent progress. In *Proceedings of the Symposium on Computing and Statistics*, Montreal, Canada, April 17–20.
26. Vapnik, V. (1995). *The Nature of the statistical learning theory*. Springer Verlag, New York.
27. Wilfong, G. (1992). Nearest neighbor problems. *International Journal of Computational Geometry & Applications*, 2(4):383–416.
28. Martinez, T.R., & Wilson, D.R. (2000). Reduction techniques for instance-based learning algorithms. *Machine Learning*, 38(3):257–286.

Multi-objective Model Predictive Optimization using Computational Intelligence

Hirotaka Nakayama and Yeboon Yun

Abstract In many engineering design problems, the explicit function form of objectives/constraints can not be given in terms of design variables. Given the value of design variables, under this circumstance, the value of those functions is obtained by some simulation analysis or experiments, which are often expensive in practice. In order to make the number of analyses as few as possible, techniques for model predictive optimization (also referred to as sequential approximate optimization or metamodeling) which make optimization in parallel with model prediction have been developed. In this paper, we discuss several methods using computational intelligence for this purpose along with applications to multi-objective optimization under static/dynamic environment.

1 Brief Review of Model Predictive Methods

To begin with, we shall review several typical methods for model prediction. Response Surface Method (RSM) has been probably most widely applied to our aim [6]. The role of RSM is to predict the response y for the vector of design variables $\boldsymbol{x} = (x_1, \ldots, x_n)$ on the basis of the given sampled observations $(\tilde{\boldsymbol{x}}_i, \tilde{y}_i), i = 1, \ldots, \ell$.

Usually, Response Surface Method is a generic name, and it covers a wide range of methods. Above all, methods using design of experiments are famous. However, many of them use relatively low order (say, 1st or 2nd) polynomials on the basis of statistical analysis in design variable space. They may provide a good approximation

Hirotaka Nakayama
Department of Info. Sci. & Sys. Eng., Konan University, Kobe 658-8501, Japan, e-mail: nakayama@konan-u.ac.jp

Yeboon Yun
Department of Reliability-based Information Systems Engineering, Kagawa University,Takamatsu 761-0396, Japan, e-mail: yun@eng.kagawa-u.ac.jp

Please use the following format when citing this chapter:

Nakayama, H. and Yun, Y., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 319328.

of black-box functions with a mild nonlinearity. It is clear, however, that in cases in which the black-box function is highly nonlinear, we can obtain better performance by methods using computational intelligence such as RBFN (Radial Basis Function Networks) or SVR (Support Vector Regression) taking into account not only the statistical property in design variable space but also that of range space of the black-box function (in other words, the shape of function).

In design of experiments, for example, D-optimality may be used for selecting a new additional sample to minimize the variance covariance matrix of the least squared error prediction. With the design matrix $\boldsymbol{X}$, this reduces to minimize the matrix $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ which is attained by maximizing $\det(\boldsymbol{X}^T\boldsymbol{X})$. This is the idea of D-optimality in design of experiments.

Other criteria are possible: to minimize the trace of $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ (A-optimality), to minimize the maximal value of the diagonal components of $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ (minimax criterion), to maximize the minimal eigen value of X^TX (E-optimality). In general, D-optimality criterion is widely used for many practical problems.

Jones *et al.* [5] suggested a method called EGO (Efficient Global Optimization) for black-box objective functions. They applied a stochastic process model for predictor and the expected improvement as a figure of merit for additional sample points. Regard y as a realized value of the stochastic variable Y, and let $f^p_{\min}$ be the minimal value of p-samples which are evaluated already. For minimization cases, the improvement at $\boldsymbol{x}$ is $I = \max\bigl(f^p_{\min} - Y,\ 0\bigr)$. Therefore, the expected improvement is given by

$$\mathrm{E}[I(\boldsymbol{x})] = \mathrm{E}\left[\max\bigl(f^p_{\min} - Y,\ 0\bigr)\right].$$

We select a new sample point which maximizes the expected improvement. Although Jones *et al.* proposed a method for maximizing the expected improvement by using the branch and bound method, we can select the best one among several candidates which are generated randomly in the design variable space. It has been observed through our experiences that this method is time consuming.

2 Using Computational Intelligence

Recently, the authors proposed to apply machine learning techniques such as RBF (Radial Basis Function) networks and Support Vector Machines (SVM) for approximating the black-box function [7], [8]. There, additional sample points are selected by considering both global and local information of the black-box function.

Support vector machine (SVM) has been recognized as a powerful machine learning technique. SVM was originally developed for pattern classification and later extended to regression ([1], [13]). In pattern classification problems with two class sets, it generalizes linear classifiers into high dimensional feature spaces through nonlinear mappings defined implicitly by kernels in the Hilbert space so that it may produce nonlinear classifiers in the original data space. Linear classifiers then are optimized to give the maximal margin separation between the classes.

This task is performed by solving some type of mathematical programming such as quadratic programming (QP) or linear programming (LP).

Linear classifiers on the basis of goal programming, on the other hand, were developed extensively in 1980's [3], [4]. The authors developed several varieties of SVM using multi-objective programming and goal programming (MOP/GP) techniques [10]. In the goal programming approach to linear classifiers, we consider two kinds of deviations: One is the exterior deviation ξ_i which is a deviation from the hyperplane of a point $\boldsymbol{x}_i$ improperly classified; The other one is the interior deviation η_i which is a deviation from the hyperplane of a point $\boldsymbol{x}_i$ properly classified. Several kinds of objective functions are possible in this approach as follows:

i) minimize the maximum exterior deviation (decrease errors as much as possible),

ii) maximize the minimum interior deviation (i.e., maximize the margin),

iii) maximize the weighted sum of interior deviation,

iv) minimize the weighted sum of exterior deviation.

Introducing the objective iv) above leads to the soft margin SVM with slack variables (or, exterior deviations) ξ_i $(i = 1, \ldots, \ell)$ which allow classification errors to some extent.

Taking into account the objectives (ii) and (iv), we can have the same formulation of ν-support vector algorithm developed by Schölkopf et al. [12]. Although many variants are possible, $\mu-\nu-$SVM considering the objectives i) and ii) is promising, because $\mu-\nu-$SVM for regression has been observed to provide a good sparse approximation [10].

The primal formulation of $\mu-\nu-$SVR is given by

$$\begin{aligned} \underset{\boldsymbol{w},b,\varepsilon,\xi,\acute{\xi}}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \nu\varepsilon + \mu(\xi + \acute{\xi}) \\ \text{subject to} \quad & \left(\boldsymbol{w}^T\boldsymbol{z}_i + b\right) - y_i \leqq \varepsilon + \xi, \;\; i = 1, \ldots, \ell, \\ & y_i - \left(\boldsymbol{w}^T\boldsymbol{z}_i + b\right) \leqq \varepsilon + \acute{\xi}, \;\; i = 1, \ldots, \ell, \\ & \varepsilon, \;\; \xi, \;\; \acute{\xi} \geqq 0, \end{aligned}$$

where ν and μ are trade-off parameters between the norm of $\boldsymbol{w}$ and ε and ξ $(\acute{\xi})$.

Applying the Lagrange duality theory, we obtain the following dual formulation of $\mu-\nu-$SVR:

$$\begin{aligned}
\underset{\alpha_i,\acute{\alpha}_i}{\text{maximize}} \quad & -\frac{1}{2}\sum_{i,j=1}^{\ell}\left(\acute{\alpha}_i-\alpha_i\right)\left(\acute{\alpha}_j-\alpha_j\right)K\left(\boldsymbol{x}_i,\boldsymbol{x}_j\right)+\sum_{i=1}^{\ell}\left(\acute{\alpha}_i-\alpha_i\right)y_i \\
\text{subject to} \quad & \sum_{i=1}^{\ell}\left(\acute{\alpha}_i-\alpha_i\right)=0, \\
& \sum_{i=1}^{\ell}\acute{\alpha}_i \leqq \mu, \quad \sum_{i=1}^{\ell}\alpha_i \leqq \mu, \\
& \sum_{i=1}^{\ell}\left(\acute{\alpha}_i+\alpha_i\right) \leqq \nu, \\
& \acute{\alpha}_i \geqq 0, \quad \alpha_i \geqq 0, \quad i=1,\ldots,\ell.
\end{aligned}$$

It has been observed through our experiences that $\mu-\nu-$SVR provides the least number of support vectors among existing SVRs. This implies that $\mu-\nu-$SVR can be effectively applied for selecting a new sample on the basis of information of support vector.

3 Using global and local information for adding new samples

If the current solution is not satisfactory, namely if our stopping condition is not satisfied, we need some additional samples in order to improve the approximation of the black-box objective function.

If the current optimal point is taken as such additional data, the estimated optimal point tends to converge to a local maximum (or minimum) point. This is due to lack of global information in predicting the objective function.

On the other hand, if additional data are taken far away from the existing data, it is difficult to obtain more detailed information near the optimal point. Therefore, it is hard to obtain a solution with a high precision. This is because of insufficient information near the optimal point.

It is important to get well balanced samples providing both global information and local information on black-box objective functions. The author and his coresearchers suggested a method which gives both global information for predicting the objective function and local information near the optimal point at the same time [7]. Namely, two kinds of additional samples are taken simultaneously for relearning the form of the objective function. One of them is selected from a neighborhood of the current optimal point in order to add local information near the (estimated) optimal point. The size of this neighborhood is controlled during the convergence process. The other one is selected far away from the current optimal value in order to give a better prediction of the form of the objective function. The former additional data gives more detailed information near the current optimal point. The latter data prevents converging to local maximum (or minimum) point.

4 Multi-objective Model Predictive Optimization: Static Cases

In multi-objective optimization, the so-called Pareto solution is introduced. Since there may be many Pareto solutions in practice, the final decision should be made among them taking the total balance over all criteria into account. This is a problem of value judgment of DM. The totally balancing over criteria is usually called *trade-off*. Interactive multi-objective programming searches a solution in an interactive way with DM while making trade-off analysis on the basis of DM's value judgment. Among them, the aspiration level approach is now recognized to be effective in practice. As one of aspiration level approaches, one of authors proposed the satisficing trade-off method [9]. Suppose that we have objective functions $\boldsymbol{f}(\boldsymbol{x}) := (f_1(\boldsymbol{x}), \ldots, f_r(\boldsymbol{x}))$ to be minimized over $\boldsymbol{x} \in X \subset R^n$. In the satisficing trade-off method, the aspiration level at the k-th iteration $\overline{\boldsymbol{f}}^k$ is modified as follows:

$$\overline{\boldsymbol{f}}^{k+1} = T \circ P(\overline{\boldsymbol{f}}^k).$$

Here, the operator P selects the Pareto solution nearest in some sense to the given aspiration level $\overline{\boldsymbol{f}}^k$. The operator T is the trade-off operator which changes the k-th aspiration level $\overline{\boldsymbol{f}}^k$ if DM does not compromise with the shown solution $P(\overline{\boldsymbol{f}}^k)$. Of course, since $P(\overline{\boldsymbol{f}}^k)$ is a Pareto solution, there exists no feasible solution which makes all criteria better than $P(\overline{\boldsymbol{f}}^k)$, and thus DM has to trade-off among criteria if he wants to improve some of criteria. Based on this trade-off, a new aspiration level is decided as $T \circ P(\overline{\boldsymbol{f}}^k)$. Similar process is continued until DM obtains an agreeable solution.

The operation which gives a Pareto solution $P(\overline{\boldsymbol{f}}^k)$ nearest to $\overline{\boldsymbol{f}}^k$ is performed by some auxiliary scalar optimization:

$$\max_{1 \leqq i \leqq r} \omega_i \left(f_i(\boldsymbol{x}) - \overline{f}_i \right) + \alpha \sum_{i=1}^{r} \omega_i f_i(\boldsymbol{x}), \tag{1}$$

where α is usually set a sufficiently small positive number, say 10^{-6}.

The weight ω_i is usually given as follows: Let f_i^* be an ideal value which is usually given in such a way that $f_i^* < \min \{ f_i(\boldsymbol{x}) \mid \boldsymbol{x} \in X \}$. For this circumstance, we set

$$\omega_i^k = \frac{1}{\overline{f}_i^k - f_i^*}. \tag{2}$$

Now, we propose a method combining the satisficing trade-off method for interactive multi-objective programming and the sequential approximate optimization using $\mu-\nu-$SVR. In the following, we explain the method along an example of the welded beam design problem [2] shown by Fig. 1. The problem is formulated as follows:

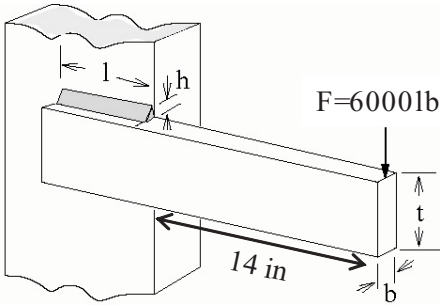


Fig. 1 Welded beam design problem

$$
\begin{aligned}
\underset{h,l,t,b}{\text{minimize}} \quad & f_1 := 1.10471h^2l + 0.04811tb(14+l) \\
\underset{h,l,t,b}{\text{minimize}} \quad & f_2 := \frac{2.1952}{t^3b} \\
\text{subject to} \quad & g_1 := \tau \leqq 13600 \\
& g_2 := \sigma \leqq 30000 \\
& g_3 := h - b \leqq 0 \\
& g_4 := P_c \geqq 6000 \\
& 0.125 \leqq h,\ b \leqq 5.0,\ \ 0.1 \leqq l,\ t \leqq 10.0
\end{aligned}
$$

Here,

$$
\begin{aligned}
\tau &= \sqrt{(\tau')^2 + (\tau'')^2 + \frac{l\tau'\tau''}{\sqrt{0.25(l^2+(h+t)^2)}}} \\
\tau' &= \frac{6000}{\sqrt{2}hl} \\
\tau'' &= \frac{6000(14+0.5l)\sqrt{0.25(l^2+(h+t)^2)}}{\sqrt{2}hl\left(\frac{l^2}{12}+0.25(h+t)^2\right)} \\
\sigma &= \frac{504000}{t^2b},\ P_c = 64746.022(1-0.0282346t)tb^3
\end{aligned}
$$

The ideal value and aspiration level are given s follows:

$$
\begin{aligned}
\text{ideal value} &:= (f_1^*, f_2^*) = (0,\ 0) \\
\text{aspiration level 1} &:= (\overline{f}_1^1, \overline{f}_2^1) = (4,\ 0.003) \\
\text{aspiration level 2} &:= (\overline{f}_1^2, \overline{f}_2^2) = (20,\ 0.002) \\
\text{aspiration level 3} &:= (\overline{f}_1^3, \overline{f}_2^3) = (40,\ 0.0002)
\end{aligned}
$$

Table 1 Result by SQP using a quasi-Newton method without model prediction

		h	l	t	b	f_1	f_2	# evaluation
asp.	average	0.5697	1.7349	10	0.5804	5.0102	3.78E-03	249.9
level	stdv	0.0409	0.1826	0	0.0072	0.0420	4.83E-05	69.6
1	max	0.5826	2.2546	10	0.5826	5.0235	3.92E-03	369.0
	min	0.4533	1.6772	10	0.5599	4.8905	3.77E-03	164.0
asp.	average	1.0834	0.8710	10.0000	1.7685	13.7068	1.25E-03	204.2
level	stdv	0.3274	0.1662	5.11E-08	0.1828	1.3793	1.13E-04	30.1
2	max	2.0132	0.9896	10	2.1263	16.3832	1.31E-03	263.0
	min	0.9221	0.4026	10.0000	1.6818	13.0527	1.03E-03	172.0
asp.	average	1.7345	0.4790	10	5	36.4212	4.39E-04	251.9
level	stdv	0.0000	0.0000	0	0	0.0000	5.71E-20	146.2
3	max	1.7345	0.4790	10	5	36.4212	4.39E-04	594.0
	min	1.7345	0.4790	10	5	36.4212	4.39E-04	112.0

Table 2 Result by the proposed method with 100 evaluations of function

		h	l	t	b	f_1	f_2
asp.	average	0.5223	1.9217	9.9934	0.5825	5.0344	3.78E-03
level	stdv	0.0374	0.1656	0.0136	0.0011	0.0130	1.08E-05
1	max	0.5832	2.2742	10	0.5845	5.0692	3.81E-03
	min	0.4520	1.6859	9.9558	0.5817	5.0224	3.77E-03
asp.	average	0.8921	1.0398	9.9989	1.6809	13.0653	1.31E-03
level	stdv	0.0898	0.1106	0.0012	0.0012	0.0081	7.79E-07
2	max	1.0787	1.1895	10	1.6824	13.0781	1.31E-03
	min	0.7849	0.8273	9.9964	1.6789	13.0531	1.31E-03
asp.	average	2.2090	0.4486	10	5	36.6830	4.39E-04
level	stdv	0.9355	0.2293	0	0	0.2695	5.71E-20
3	max	3.7812	0.8734	10	5	37.1257	4.39E-04
	min	1.0391	0.1895	10	5	36.4212	4.39E-04

Table 1 shows the result by the simple satisficing trade-off method using SQP and a quasi-Newton method for randomly chosen starting points in 10 times. Table 2 shows the result by our proposed method combining the satisficing trade-off method and the model predictive optimization using μ-ν-SVR with 100 sample points. Since we used the usual gradient based optimization method for the simple satisficing trade-off method, the number of function evaluation would be almost 4 times for black box functions because we have to apply the numerical differentiation on the based on the incremental difference.

5 Multi-objective Model Predictive Optimization: Dynamic Cases

For dynamic optimization problems, the model predictive control has been developed along a similar idea to the above. Let $\boldsymbol{u}(t),\ \boldsymbol{x}(t)$ denote the control (input) vector and the state vector, respectively. Our problem is represented by

$$\text{Minimize} \qquad J = \phi[\boldsymbol{x}(T)] + \int_0^T F(\boldsymbol{x}, \boldsymbol{u}, t)dt \tag{3}$$

$$\text{subject to} \qquad \dot{\boldsymbol{x}} = f(\boldsymbol{x}(t), \boldsymbol{u}(t), t),\ \ \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{4}$$

If the function form in the above model is explicitly given, then we can apply some techniques in optimal control theory. However, we assume that some of function forms, in particular the dynamic system equation (4), can not explicitly be given. Under this circumstance, we predict some of future state $\boldsymbol{x}(t+1), \ldots, \boldsymbol{x}(t+k)$ for given $\boldsymbol{u}(t+1), \ldots, \boldsymbol{u}(t+p)$. The period $[t+1, t+k]$ is called the prediction period, and $[t+1, t+p]$ the control period. Our aim is to decide the optimal control sequence $\boldsymbol{u}(t)$ over $[0, T]$.

For predicting the future state, we apply a support vector regression technique, namely $\mu - \nu-$SVR which was stated in the previous section. In the following, we show a numerical result by using the satisficed trade-off method with model prediction. Our problem to be considered in this paper has multiple objectives

$$J = (J_1, \ldots, J_r).$$

For example, those objectives are the energy consumption, constraints of terminal state, the terminal time (T) itself and so on.

Step 1. Predict the model f based on $(x(k), u(k), x(k+1)),\ \ k = 0, 1, \ldots, t-1,\ x(0) = x_0$.

Step 2. Generate individuals of control sequence by GA algorithm

$$u_i(t),\ u_i(t+1), \ldots, u_i(T-1),\ \ i = 1, 2, \ldots, N_{population}.$$

- Predict the state resulting from each control sequence from the present time to the terminal time

$$x(k+1) = f(x(k), u(k)),\ k = t, t+1, \ldots, T-1,\ \ x(0) = x_0. \tag{5}$$

- Evaluate each individual in terms of auxiliary scalar objective function of satisficing trade-off method

$$F = \max_{1 \leqq i \leqq r} w_i \left(J_i(x) - \overline{J}_i^k \right) + \alpha \sum_{i=1}^{r} w_i \left(J_i(x) - \overline{J}_i^k \right)$$

$$J_1 = T,\ J_2 = \sum_{k=t}^{T-1} u^2(k).$$

- Select the best individual (control sequence) u^*. Calculate $x(t+1)$ by (5) using $x(t)$ and $u(t) = u^*(t)$.

Step 3. $t \leftarrow t + 1$ and go to Step 2.

The solutions for two different aspiration levels are depicted in Fig. 2. It may be seen that the proposed method provides reasonable solutions flexibly depending on the apiration levels.

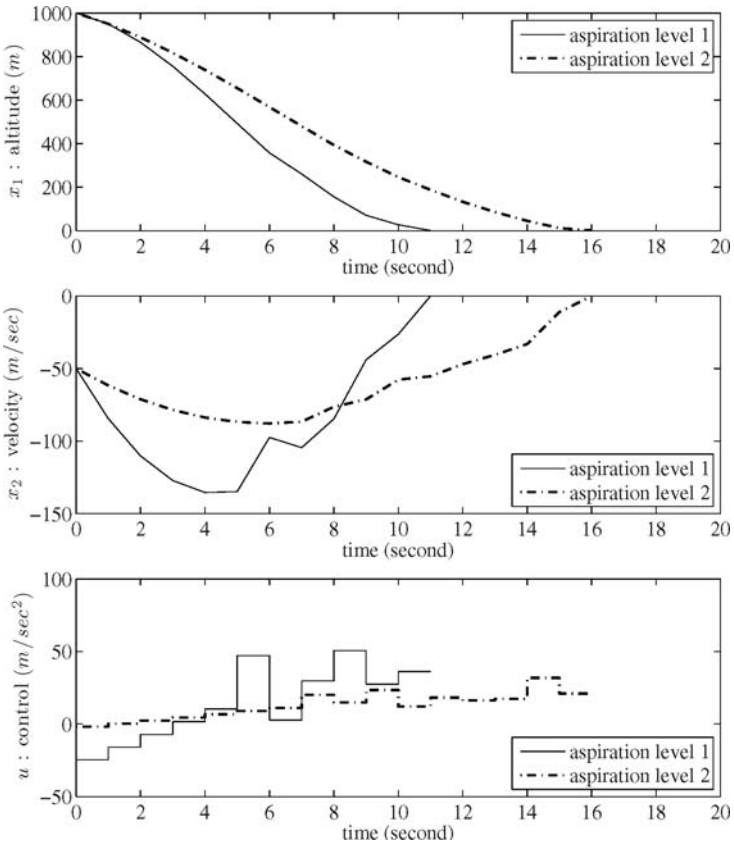


Fig. 2 Multi-objective model predictive control

6 Concluding Remarks

We discussed methods combining the satisficing trade-off method and model predictive optimization methods using computational intelligence under static and dynamic environment. The proposed method provides an approximate Pareto solution closest to the given aspiration level. It is promising in practical problems since it has been observed through several numerical experiments that the method reduces the number of function evaluation up to less than $1/100$ to $1/10$ of usual methods.

References

1. Cortes, C. and Vapnik, V., (1995) Support Vector Networks, *Machine Learning*, **20**, pp. 273-297
2. Deb, K. and Srinivasan, A., (2005) Innovization:Innovative Design Principles Through Optimization, KanGAL Report #2005007, Indian Institute of Technology
3. Erenguc, S.S. and Koehler, G.J., (1990) Survey of Mathematical Programming Models and Experimental Results for Linear Discriminant Analysis, *Managerial and Decision Economics*, **11**, 215-225
4. Freed, N. and Glover, F., (1981) Simple but Powerful Goal Programming Models for Discriminant Problems, *European J. of Operational Research*, **7**, 44-60
5. Jones, D.R., Schonlau, M. and Welch, W.J., (1998) Efficient Global Optimization of Expensive Black-Box Functions, *J. of Global Optimization*, **13**, 455-92
6. Myers, R.H. and Montgomery, D.C., (1995) *Response Surface Methodology: Process and Product Optimization using Designed Experiments*, Wiley
7. Nakayama, H., Arakawa, M. and Sasaki, R., (2002) Simulation based Optimization for Unknown Objective Functions, *Optimization and Engineering*, **3**, 201-214
8. Nakayama, H., Arakawa, M. and Washino, K., (2003) Optimization for Black-box Objective Functions, *Optimization and Optimal Control*, (eds.) P.M. Pardalos, I. Tseveendorj and R. Enkhbat, World Scientific, 185-210
9. Nakayama, H. and Sawaragi, Y., (1984), Satisficing Trade-off Method for Multi- objective Programming, in M. Grauer and A. Wierzbicki (eds.): *Interactive Decision Analysis*, Springer, 113-122
10. Nakayama, H. and Yun, Y., (2006) Generating Support Vector Machines using Multiobjective Optimization and Goal Programming *Multi-objective Machine Learning*, Yaochu Jin(ed.), Springer Series on Studies in Computational Intelligence, pp.173-198.
11. Nakayama, H. and Yun, Y., (2006) Support Vector Regression Based on Goal Programming and Multi-objective Programming, IEEE World Congress on Computational Intelligence, CD-ROM (Paper ID : 1536)
12. Schölkopf, B. and Smola, A.J., (1998) New Support Vector Algorithms, *NeuroCOLT2 Technical report Series*, NC2-TR-1998-031
13. Vapnik, V.N., (1998) *Statistical Learning Theory*, John Wiley & Sons, New York
14. Yoon, M., Yun, Y.B. and Nakayama, H., (2003) A Role of Total Margin in Support Vector Machines, *Proc. IJCNN'03*, 2049-2053

An Intelligent Method for Edge Detection based on Nonlinear Diffusion

C. A. Z. Barcelos and V. B. Pires

Abstract Edge detection is an important task in the field of image processing with broad applications in image and vision analysis. In this paper, we present a new intelligent computational mechanism using nonlinear diffusion equations for edge detection. Experimental results show that the proposed method outperforms standard edge detectors as well as other methods that deploy inhibition of texture.

1 Introduction

The conventional edge detectors as the Canny edge detector and others [5, 8, 17], do not make distinction between isolated edges and edges originating from texture. Therefore many false edges usually deriving from textures and noises are detected by these algorithms.

Significant advances have been achieved with the use of more sophisticated techniques such as the algorithms based on nonlinear diffusion [1, 3, 13] and inspired by the human visual system (HSV) [7, 9, 12], among others.

In this paper, we present a new intelligent computational method to effectively edge detect in natural images. We incorporate a nonlinear diffusion equation to the Canny edge detector, and show that this results in a method more effective to edges detection in presence of texture. The proposed method can be divided into two stages. The first consists of the application of a nonlinear diffusion equation, whose main idea is to accomplish, the selective smoothing of the image of interest, removing the irrelevant information, usually related to noise and texture elements. The

C. A. Z. Barcelos
Federal University of Uberlndia, FAMAT/FACOM, CEP 38400-902, Brazil
e-mail: celiazb@ufu.br

V. B. Pires
Federal University of Uberlndia, FACOM, CEP 38400-902, Brazil
e-mail: vinniciusbp@yahoo.com.br

Please use the following format when citing this chapter:

Barcelos, C.A.Z. and Pires, V.B., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 329338.

second stage consists of the application of the Canny edge detector to the smoothed image, in an attempt to obtain as a final result only those edges of interest [14].

The proposed method is able to minimize the inconvenience effect of false edge detection (usually deriving from textures and noises) and improve the performance of the traditional Canny edge detector. Results obtained from various natural images are used to demonstrate the performance of proposed method. Comparing our new method with other edge detection methods, better performance in terms of removing texture in the edge detection process was obtained.

This paper is organized as follows: section 2 describes the proposed edge detection method. The experimental results that exemplify the performance of the proposed method and the numerical implementation of the method are described in section 3, and finally section 4 presents the paper's conclusion.

2 Proposed Edge Detection Method

Motivated by the difficulty found by standard edge detectors to remove noises and textures, we proposed an intelligent computational method for edge detection in digital images adding the nonlinear diffusion method introduced in [3] to the Canny edge detector [6].

The proposed method for edge detection can be divided into two stages. The first stage consists of the application of a nonlinear diffusion equation for selective smoothing of the image of interest, in an attempt of minimize the false edge detection originating from noise and irrelevant features. In the second stage, our goal is to apply the Canny edge detector in fine scale Gaussian on the smoothed image, since noises and texture elements were effectively removed by smoothing process of the considered diffusion model, in an attempt to obtain a map with its edges refined as a final result.

2.1 Edges Detection via Nonlinear Diffusion Equations

Several smoothing methods can be found in literature, however undesirable effects such as edge deterioration, loss of relevant information, make some of these methods unviable when one desires to eliminate just irrelevant information such as noise and at the same time maintain the edges intact.

During the last few years, many mathematical models have been proposed in the attempt to solve these problems related to image smoothing. We can cite, for example [1, 3, 10, 13] which are briefly described as follows.

Perona and Malik [13] developed a model through an anisotropic diffusion equation in the attempt to preserve the exact location of an edge point.

The mathematical formulation of the model is given by:

$$u_t = div(g(|\nabla u|)\nabla u), \quad x \in \Omega, t > 0, \tag{1}$$
$$u(x,0) = I(x), \quad x \in \Omega \subset R^2,$$

where $I(x)$ is the original image, $u(x,t)$ is its smoothed version on the scale t, g is a smooth non-increasing function, such that $g(0) = 1$, $g(s) \geq 0$ and $g(s) \to 0$ when $s \to \infty$. The usual choice for g is given by $g(s) = \frac{1}{1+ks^2}$ where k is a parameter.

This model provides edges and contours that remain stable through the scale t. However, this model presents many theoretical and practical difficulties. For example, if the image is very noisy, the "size" of the gradient ∇u will be very large at almost all of the points on the image and, as a consequence the function g will be almost null at these same points. In this way, all the noise on the image remains even when the image is processed by the smoothing process introduced by this model.

As this equation is not able to preserve the edges localization, Alvarez, Lions and Morel proposed in [1] the following nonlinear parabolic equation:

$$u_t = g(|\nabla G_\sigma * u|)|\nabla u| \; div\left(\frac{\nabla u}{|\nabla u|}\right), \quad x \in \Omega, t > 0, \tag{2}$$
$$u(x,0) = I(x), \quad x \in \Omega \subset R^2,$$

where $u(x,0) = I(x)$ is the initial image, $\frac{\partial u}{\partial \eta}\big|_{\partial\Omega \times R_+} = 0$ is the boundary condition and $g(|\nabla G_\sigma * u|) = \frac{1}{1+k|\nabla G_\sigma * u|^2}$, with $k > 0$ being a parameter and G_σ being the Gaussian function.

This model dislocates the level curves of the image u in the orthogonal direction ∇u with speed $g(|\nabla G_\sigma * u|)K$, where $K = div(\frac{\nabla u}{|\nabla u|})$ is the local curvature of the iso-intensity contour. Therefore, the image is smoothed along both sides of the edges with minimal smoothing of the edge itself, and this is carried out at speeds which are lower near the edges and higher in the interior of homogeneous regions which makes the preservation of the edges of the image possible.

Another modification in the Perona and Malik model Eq.(1) was proposed by Nordström [10]. He added a forcing term $(u - I)$ to the Perona and Malik model Eq.(1) forcing $u(x,t)$ to maintain itself close to the initial image $I(x)$.

The presence of the forcing term in Eqs.(1) and (2) reduces the degenerative effects of the diffusion to very acceptable levels; however, the models with this forcing term do not eliminate noise satisfactorily [3].

In the attempt to obtain ever better results, Barcelos, Boaventura and Silva Jr. [3] also presented a mathematical model for the smoothing and segmentation of digital images through a diffusion equation given by:

$$u_t = g\,|\nabla u|\, div\left(\frac{\nabla u}{|\nabla u|}\right) - \lambda\,(1-g)(u-I), \quad x \in \Omega, t > 0, \tag{3}$$
$$u(x,0) = I(x), \quad x \in \Omega \subset R^2,$$

$$\frac{\partial u}{\partial \eta}\Big|_{\partial\Omega\times R_+} = 0, \quad x \in \partial\Omega, t > 0,$$

where $g = g(|\nabla G_\sigma * u|)$, $I(x)$ is the original image, $u(x,t)$ is its smoothed version on the scale t, λ is a parameter, $(u-I)$ is the term suggested by Nordström and $(1-g)$ is the moderation selector introduced in this model.

The balanced diffusion of the image allows the homogeneous regions $g \sim 1$ to be smoothed even more in relation to the edge regions $g \sim 0$. This is obtained through the moderation selector $(1-g)$ which by being in function with g allows for the identification of these different regions on the image [3].

Experimental results obtained with the application of this model show its efficiency in the smoothing process of natural images.

2.2 Canny Edges Detector

Among the edge detection methods found in literature, the Canny edge detector is considered to be one of the most used algorithm for edge detection.

The Canny edge detection process is based upon three basic performance criteria: good detection, good localization, and single response [2, 6]. The main objective of Canny work is to develop an optimal detector.

The implementation of the Canny edge detector [6] follows the steps below.

First, the input image $I(x)$ is smoothed to remove irrelevant details like noises and texture. The smoothing is obtained by convolution of the image $I(x)$ with a Gaussian function G_σ.

Second, determine gradient magnitude $|\nabla I(x)|$ and gradient direction at each pixel (x) in the smoothed image.

In the third step, non-maxima suppression technique is performed. In this process, all the pixels (x) for which the gradient magnitude $|\nabla I(x)|$ has a local maximum in the gradient direction will be considered edge pixels.

Fourth, hysteretic thresholding is performed to remove the weak edges. In this process, two different thresholds are used: the low threshold t_L and the high threshold t_H. All candidate edge pixels with the gradient magnitude below the low threshold t_L are considered as non edges. Only the pixels with the gradient magnitude above the low threshold t_L that can be connected to any pixel with magnitude above the high threshold t_H are considered as edge pixels.

3 Computational Details and Experimental Results

Obtaining the edges of natural images, such as animals in their natural habitat or objects on textured background, is not an easy task. The more sophisticated edge

detection algorithms try to find an edge map that approximates the ideal edge map, which is usually drawn by hand more closely.

In the paper, we use natural images with different complexity levels and their corresponding ground truth maps to evaluate the performance of the proposed edge detection method. The natural images and ground truth contour maps were obtained in http://www.cs.rug.nl/ imaging/papari/JASP/results.html.

3.1 Performance Measure

To compare our method with the the Canny edge detector [6] and the single scale surround inhibition algorithm [9], we use the performance measure introduced in [9].

Let DO be the number of correctly detected edge pixels, FP the number of false positive pixels, i.e. pixels considered as edges by the detector while they belong to the background of the desired output and FN the number of false negative pixels, i.e. desired output edge pixels missed by the operator. The performance measure introduced in [9] is defined as:

$$P = \frac{DO}{DO + FP + FN} \tag{4}$$

Note that if $FN = 0$, i.e. all true edge pixels are correctly detected and if $FP = 0$, i.e. no background pixels are falsely detected as edge pixels, then $P = 1$. On the other hand, if FP (edge pixels falsely detected) and/or FN (true edge pixels missed by the detector) are greater, the value of P will be lower.

For the implementation of the above measure, we consider that an edge pixel is correctly detected if a corresponding ground truth edge pixel is present in a 5×5 square neighborhood centered at the respective pixel coordinates.

3.2 Numerical Implementation

Numerical solution of the mathematical model Eq.(3) is obtained by applying appropriate finite difference methods [11, 15, 16].

The images are represented by NxM matrices of intensity values. Let u_{ij} denote the value of the intensity of the image u at the pixel (x_i, y_j) with i = 1, 2,..., N and j = 1, 2,..., M. The evolution equation obtains images at times $t_n = n\Delta t$ where Δt is the step time and n = 1, 2,... . We denote $u(x_i, y_j, t_n)$ by u_{ij}^n.

Let

$$\mathscr{L}(u) = g\,|\nabla u|\, div\left(\frac{\nabla u}{|\nabla u|}\right) - \lambda\,(1-g)(u-I). \tag{5}$$

We can write Eq.(3) in the form $u_t = \mathscr{L}(u)$. The time derivative u_t at (i, j, t_n) is approximated using the Euler method $\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t}$, so the discretization of the Eq.(3) is given by

$$u_{ij}^{n+1} = u_{ij}^n + \Delta t \mathscr{L}(u_{ij}^n)$$

where $u_{ij}^0 = I(x_i, y_i)$.

The diffusion term

$$|\nabla u| \, div\left(\frac{\nabla u}{|\nabla u|}\right) = \frac{u_x^2 u_{yy} - 2u_x u_y u_{xy} + u_y^2 u_{xx}}{u_x^2 + u_y^2} \tag{6}$$

in Eq. (3) is approximated using central differences, i.e:

$$u_x(x_i, y_j) \approx \frac{u_{i+1,j} - u_{i-1,j}}{2h},$$

$$u_y(x_i, y_j) \approx \frac{u_{i,j+1} - u_{i,j-1}}{2h},$$

$$u_{xx}(x_i, y_j) \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2},$$

$$u_{yy}(x_i, y_j) \approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2},$$

$$u_{xy}(x_i, y_j) \approx \frac{1}{4h^2}\Big[u_{i+1,j+1} - u_{i+1,j-1} - u_{i-1,j+1} + u_{i-1,j-1}\Big],$$

with $i = 1, ..., N$ and $j = 1, ..., M$.

The function g is given by

$$g = \frac{1}{1 + k|\nabla G_\sigma * u|^2} \tag{7}$$

where k is a parameter and G_σ is given by

$$G_\sigma(x, y) = \frac{1}{\sigma\sqrt{2\pi}} \, e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

3.3 Results

Here we present some results obtained with the application of the proposed edge detection method in several natural images with different complexity levels. We

evaluate the performance P of proposed edge detector, and compare it to the performance of two other edge detection algorithms: the Canny edge detector [6] and the single scale edge detector with surround inhibition proposed in [9].

The obtained results using 5 test images are shown in Fig. 1. The first column shows the original images while the second column shows the ground truth. The third and fourth column show the results obtained by the proposed edge detection method and by the Canny edge detector [6], respectively. The fifth column shows the results obtained from the single scale surround inhibition algorithm proposed in [9]. These results can be found in http://www.cs.rug.nl/ imaging/papari/JASP/results.html and they are being used in this paper for comparison purpose only. The performance measures P concerning the three algorithms cited above are displayed at the bottom of each image.

As we can see, for all cases presented our method (third column) gives the best performance in terms of edge detection. The proposed method has the advantage of minimizing the inconvenience effect of false edge detection. On the other hand, the worst result in terms of performance is presented by the Canny edge detector (fourth column) which does not remove effectively texture elements, while the results obtained with the single scale surround inhibition algorithm (fifth column) present a significant advantage in terms of performance measurement once it uses texture suppression mechanism.

The implementation of the proposed method uses the following parameters: the used step size, Δt, for the temporal evolution of u_t was fixed at $\Delta t = 0.25$; the constant k was chosen in a manner which allows function $g(s)$ to carry out its role, which is $g \sim 1$ when s is large (edge points) and $g \sim 1$ when s is small (interior points); the constant λ was fixed at $\lambda = 1$, this means that the balance between the smoothing and the forcing term was unweighted. In [4], the authors describe the choice of parameters with more details.

The Canny edge detector parameters used were: σ, as the standard deviation of a Gaussian derivative kernel and two thresholds t_L and t_H. In our experiments, we fixed $\sigma = 1$ and $t_L = 0.4t_H$.

Table 1 shows the parameters used in the experiments.

Test images	Parameters of the Proposed Method					
	Nonlinear Diffusion Equation				Canny	
	λ	Δ t	k	Iterations	σ	t_H
Elephant	1.0	0.25	0.003	100	1.0	0.183
Rhino	1.0	0.25	0.0015	100	1.0	0.289
Goat	1.0	0.25	0.003	100	1.0	0.391
Car	1.0	0.25	0.0007	100	1.0	0.28
Golfcart	1.0	0.25	0.0008	100	1.0	0.3

Table 1 Parameters used for each tested image.

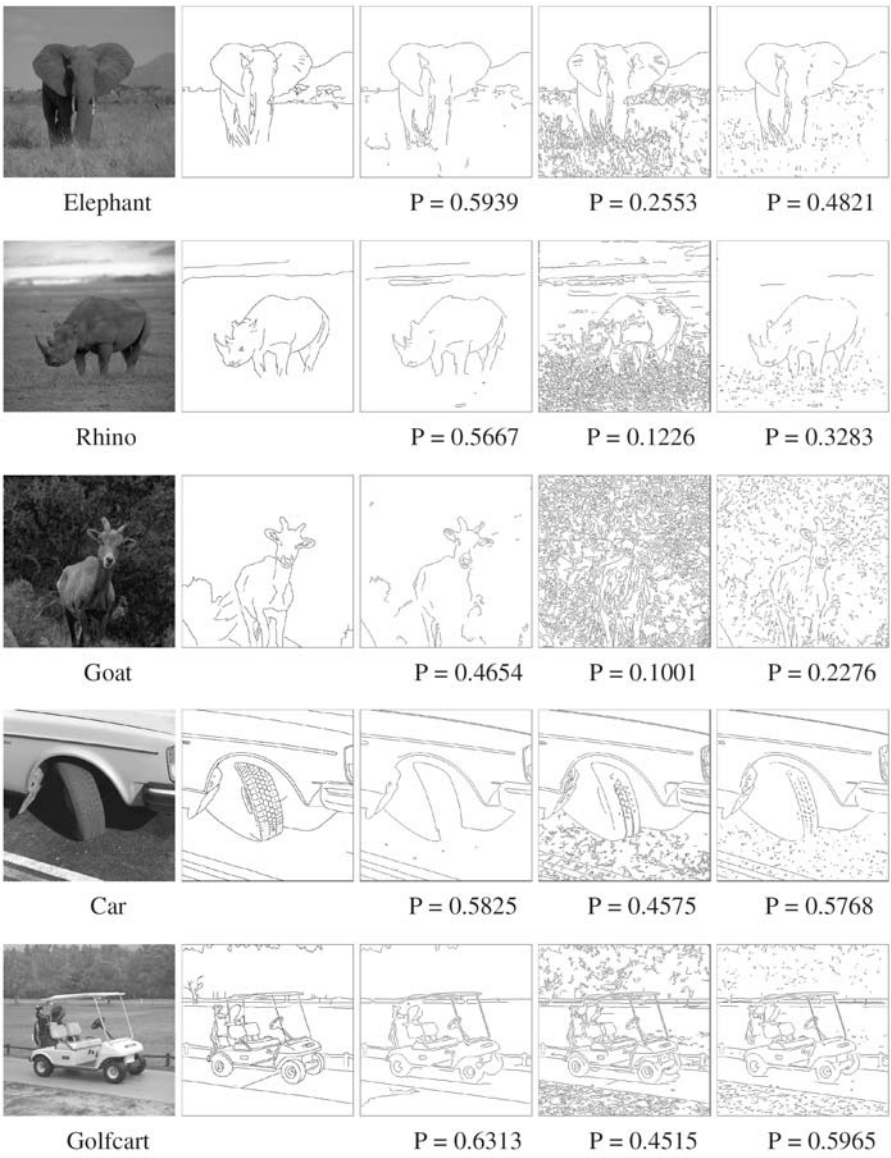


Fig. 1 Natural images (first column); Ground truth edge map (second column); Obtained results from the proposed method (third column); Obtained results from the Canny method (fourth column); Results from the single scale surround inhibition algorithm [9] (fifth column).

4 Conclusions

In this paper an edge detection method was proposed that outperforms all the considered edge detectors, even when the images background is textured.

Through the addition of the nonlinear diffusion method introduced in [3] to the Canny edge detector [6], we showed that the proposed edge detection method has the advantage of minimizing the inconvenience effect of false edge detection and at the same time to be efficient in the detection of true edges.

Due to the capacity that the nonlinear diffusion equations have to smooth an image and at the same time preserve the edges of interest, for a subsequent analysis via edge detector, we believe that the nonlinear diffusion equation introduced in [3] can also be extended the other conventional edge detectors improving the performance of same.

In summary, in this work we have shown that the proposed method is a useful computational mechanism which reflects human perception well.

Acknowledgements We acknowledge the financial support of CNPq, the Brazilian National Research Council (Grants #474406/2006-7 and #308798/2006-6) and Fapemig (CEX 2222-5).

References

1. Alvarez, L., Lions, P. L., Morel, J.M.: Image selective smoothing and edge detection by nonlinear diffusion II. SIAM journal on numerical analysis. Vol. 29, No. 3, pp. 845–866, (1992).
2. Bao, P., Zhang, L., Wu, X.: Canny edge detection enhancement by scale multiplication. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 27, No. 9, pp. 1485–1490, (2005).
3. Barcelos, C.A.Z., Boaventura, M., Silva Jr, E.C.: A well-balanced flow equation for noise removal and edge detection. IEEE Transactions on Image Processing. Vol. 12, No. 7, pp. 751–763, (2003).
4. Barcelos, C.A.Z., Boaventura, M., Silva Jr, E.C.: Edge detection and noise removal with automatic selection of paramenters foa a pde based model. Computational and Applied Mathematics. Vol. 24, No. 71, pp. 131–150, (2005).
5. Black, M., Sapiro, G., Marimont, D., Heeger, D.: Robust anisotropic diffusion. IEEE Transactions Image Processing. Vol. 7, pp. 421–432, (1998).
6. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 8, No. 6, pp. 679–698, (1986).
7. Chaji, N., Ghassemian, H.: Texture-gradient-based contour detection. EURASIP Journal on Advances in Signal Processing. Vol. (2006), pp. 1–8, (2006).
8. Demigny, D.: On optimal linear filtering for edge detection. IEEE Transactions Image Processing. Vol. 11, pp. 728–1220, (2002).
9. Grigorescu, C., Petkov, N., Westenberg, M.A.: Contour detection based on non-classical receptive field inhibition. IEEE Transactions on Image Processing. Vol. 12, No. 7, pp. 729–739, (2003).
10. Nordström, K.N.: Biased anisotropic diffusion: a unified regularization and diffusion approach to edge detection. Image and Vision Computing. Vol. 8, No. 4, pp. 318–327, (1990).
11. Osher, S., Sethian, J.: Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. Journal of Computational Physics. Vol. 79, pp. 12–49, (1988).
12. Papari, G., Campisi, P., Petkov, N., Neri, A.: A biologically motivated multiresolution approach to contour detection. EURASIP Journal on Advances in Signal Processing. Vol. 2007, pp. 1–28, (2007).
13. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 12, No. 7, pp. 629–639, (1990).

14. Pires, V.B., Barcelos, C.A.Z.: Edge detection of skin lesions using anisotropic diffusion. Seventh International Conference on Intelligent Systems Design and Applications - ISDA. Vol. 0, pp. 363–370, (2007).
15. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D. Vol. 60, pp. 259–268, (1992).
16. Sethian, J.A.: Level set methods. Cambridge University Press. (1996).
17. Shin, M.C., Goldgof, D.B., Bowyer, K.W., Nikiforou, S.: Comparison of edge detection algorithms using a structure from motion task. IEEE Transactions System, Man, and Cybernetics-Part B: Cybernetics. Vol. 31, pp. 589–601, (2001).

SEMANTIC WEB

A Survey of Exploiting WordNet in Ontology Matching

Feiyu Lin and Kurt Sandkuhl

Abstract Nowadays, many ontologies are used in industry, public adminstration and academia. Although these ontologies are developed for various purposes and domains, they often contain overlapping information. To build a collaborative semantic web, which allows data to be shared and reused across applications, enterprises, and community boundaries, it is necessary to find ways to compare, match and integrate various ontologies. Different strategies (e.g., string similarity, synonyms, structure similarity and based on instances) for determining similarity between entities are used in current ontology matching systems. Synonyms can help to solve the problem of using different terms in the ontologies for the same concept. The WordNet thesauri can support improving similarity measures. This paper provides an overview of how to apply WordNet in the ontology matching research area.

1 Introduction

The Semantic Web provides shared understanding, well structured content and reasoning for extending the current web. Ontologies are essential elements of the semantic web. Nowadays, many ontologies are used in industry, public adminstration and academia. Although these ontologies are developed for various purposes and domains, they often contain overlapping information. To build a collaborative semantic web, which allows data to be shared and reused across applications, enterprises, and community boundaries [22], it is necessary to find ways to compare, match and integrate various ontologies.

Ontology matching in general is based on finding similar entities in the source ontologies or finding translation rules between ontologies. Different strategies (e.g.,

Feiyu Lin
Jönköping University, Jönköping, Sweden, e-mail: feiyu.lin@jth.hj.se

Kurt Sandkuhl
Jönköping University, Jönköping, Sweden, e-mail: kurt.sandkuhl@jth.hj.se

Please use the following format when citing this chapter:

Lin, F. and Sandkuhl, K., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 341–350.

string similarity, synonyms, structure similarity and based on instances) for determining similarity between entities are used in current ontology matching systems. When comparing ontology entities based on their labels, synonyms can help to solve the problem of using different terms in the ontologies for the same concept. For example, an ontology might use “diagram”, another ontology could use “graph” referring to the same concern.

The WordNet[25] can support improving similarity measures. This paper provides an overview of how to apply WordNet in the ontology matching research area.

2 WordNet

WordNet is based on psycholinguistic theories to define word meaning and models not only word meaning associations but also meaning-meaning associations [7]. WordNet tries to focus on the word meanings instead of word forms, though inflection morphology is also considered. WordNet consists of three databases, one for nouns, one for verbs and a third for adjectives and adverbs. WordNet consists of a set of synonyms “synsets”. A synset denotes a concept or a sense of a group of terms. Synsets provide different semantic relationships such as synonymy (similar) and antonymy (opposite), hypernymy (superconcept)/hyponymy (subconcept)(also called Is-A hierarchy / taxonomy), meronymy (part-of) and holonymy (has-a). The semantic relations among the synsets differ depending on the grammatical category, as can be seen in Figure 1 [11]. WordNet also provides textual descriptions of the concepts (*gloss*) containing definitions and examples. WordNet can be treated as a partially ordered synonym resources.

Fig. 1 Semantic relations in WordNet. (Source: [11])

Semantic Relation	Syntactic Category	Examples
Synonymy (similar)	Noun Verb Adj Adv	pipe, tube rise, ascend sad, unhappy rapidly, speedily
Antonymy (opposite)	Adj Adv Noun Verb	wet, dry rapidly, slowly top, bottom rise, fall
Hyponymy (subordinate)	Noun	sugar maple, maple maple, tree tree, plant
Meronymy (part)	Noun	brim, hat gin, martini ship, fleet
Troponymy (manner)	Verb	march, walk whisper, speak
Entailment	Verb	drive, ride divorce, marry
Derivation	Adj Adv	magnetic, magnetism simply, simple

EuroWordNet [5] is a multilingual database with wordnets for several European languages (Dutch, Italian, Spanish, German, French, Czech and Estonian). It uses the same structure as the English WordNet. EuroWordNet can solve cross-language problems, for example, words for different languages, such as English, French, Italian, German, are used to name the same entities.

3 Exploiting WordNet in Ontology Matching

Semantic similarity based on WordNet has been widely explored in Natural Language Processing and Information Retrieval. But most of these methods are applied in an ontology (e.g., WordNet). We will first show these methods, then we will discuss how to apply them in ontology matching.

Several methods for calculating semantic similarity between words in WordNets exist and can be classified into three categories:

- Edge-based methods: to measure the semantic similarity between two words is to measure the distance (the path linking) of the words and the position of the word in the taxonomy. That means the shorter the path from one node to another, the more similar they are (e.g., [27], [18], [24]).
- Information-based statistics methods: to solve the difficult problem to find a uniform link distance in edge-based methods, Resnik proposes an information-based statistic method [19].The basic idea is that the more information two concepts have in common, the more similar they are. This approach is independent of the corpus. For examples see [19], [13].
- Hybrid methods: combine the above methods, e.g., [21], [9], [4].

3.1 Edge-based Methods

Wu and Palmer [27] propose defining the similarity of two concepts based on the common concepts by using the path.

$$sim(C_1, C_2) = \frac{2 * N_3}{N_1 + N_2 + 2 * N_3}, \tag{1}$$

where C_3 is the least common superconcept of C_1 and C_2. N_1 is the number of nodes on the path from C_1 to C_3. N_2 is the number of nodes on the path from C_2 to C_3. N_3 is the number of nodes on the path from C_3 to root.

Resnik [18] introduces a variant of the edge-based method, converting it from a distance to a similarity metric by subtracting the path length from the maximum possible path length.

$$sim_{edge}(w_1, w_2) = (2 * MAX) - [min_{c_1,c_2} len(c_1, c_2)] \tag{2}$$

where $s(w_1)$ and $s(w_2)$ represent the set of concepts in the taxonomy that are senses of word w_1, w_2 respectively, c_1 overs $s(w_1)$, c_2 overs $s(w_2)$, MAX is the maximum depth of the taxonomy, and $len(c_1, c_2)$ is the length of the shortest path from c_1 to c_2.

Su defines the similarity of two concepts based on the distance of the two concepts in WordNet [24]. This can be done by finding the paths from one concept to the other concept and then selecting the shortest such path. Threshold like 11 is set

for the top nodes of the noun taxonomy. That means not always a path can be found between two nouns. The WordNet similarity is used to adjust similarity value in his ontology matching system.

3.2 Information-based Statistics Methods

Resnik proposes an information-based statistic method [19]. First, it calculates the probability with concepts in the taxonomy, then follows information theory, the information content of a concept can be quantified as negative the log likelihood. Let $\mathscr{C}$ be set of concepts in the taxonomy. The similarity of two concepts is extent to the specific concept that subsumes them both in the taxonomy. Let the taxonomy be augmented with a function $p : \mathscr{C} \rightarrow [0,1]$, such that for any $c \in \mathscr{C}$, $p(c)$ is the probability of encountering concept c. If the taxonomy has a unique top node then its probability is 1. The information content of c can be quantified as $-\log p(c)$. Then

$$sim(c_1,c_2) = max_{c \in S(c_1,c_2)}[-\log p(c)], \tag{3}$$

where $S(c_1,c_2)$ is the set of concepts that subsume both c_1 and c_2. The word similarity (sim) is defined as

$$sim(w_1,w_2) = max_{c_1,c_2}[sim(c_1,c_2)], \tag{4}$$

where $s(w_1)$ and $s(w_2)$ represent the set of concepts in the taxonomy that are senses of word w_1, w_2 respectively, c_1 overs $s(w_1)$, c_2 overs $s(w_2)$.

Lin adapts Resnik's method and defines the similarity of two concepts as the ratio between the amount of information needed to state the commonality between them and the information needed to fully describe them [13].

$$sim(x_1,x_2) = \frac{2 \times \log p(c_0)}{\log p(c_1) + \log p(c_2)}, \tag{5}$$

where $x_1 \in c_1$ and $x_2 \in c_2$, c_0 is the most specific class that subsumes both c_1 and c_2. The ontology alignment tool RiMOM [28] includes Lin's approach in the system.

3.3 Hybrid Methods

Jiang and Conrath propose a combined model that is derived from the edge-based notion by adding the information content as a decision factor [9]. The information content $IC(c)$ of a concept c can be quantified as $-\log P(c)$. The link strength (LS) of an edge is the difference of the information content values between a child concept and its parent concept.

$$LS(c_i, p) = -\log(P(c_i|p)) = IC(c_i) - IC(p) \tag{6}$$

where child concept c_i is a subset of its parent concept p, . After considering other factors, e.g., local density, node depth, and link type, the distance function is:

$$Dist(w_1, w_2) = IC(c_1) + IC(c_2) - 2 \times IC(LSuper(c_1, c_2)), \tag{7}$$

where $LSuper(c_1, c_2)$ is the lowest super concept of c_1 and c_2.

Rodriguez presents another approach to determine similar entities based on WordNet. For example, it considers hypernym/hyponym, holonym/meronyms relations [21]. The similarity measure based on the normaliztion of Tversky's model and set theory functions (S) of intersection $|A \cap B|$ and difference $|A/B|$ is as follows:

$$S(a,b) = \frac{|A \cap B|}{|A \cap B| + \alpha(a,b)\,|A/B| + (1 - \alpha(a,b))\,|B/A|} \tag{8}$$

where a and b are entity classes, A and B are the description sets of a and b (i.e., synonym sets, is-a or part-whole relations), α is a function that defines the relative importance of the non-common characteristics. For $is-a$ hierarchy, α is expressed in term of the depth of the entity classes.

$$\alpha(a,b) = \begin{cases} \dfrac{depth(a)}{depth(a) + depth(b)} & if\ depth(a) \leq depth(b) \\ 1 - \dfrac{depth(a)}{depth(a) + depth(b)} & if\ depth(a) > depth(b) \end{cases} \tag{9}$$

Petrakis et al. adapt Rodrigues approach and develop X-Similarity which relies on synsets and term description sets [4]. Equation 8 is replaced as plain set similarity (S) where A and B mean synsets or term description sets.

$$S(a,b) = max\frac{A \cap B}{A \cup B}, \tag{10}$$

The similarity between term neighborhoods $S_{neighborhoods}$ is computed per relationship type (e.g., Is-A and Part-Of) as

$$S_{neighborhoods}(a,b) = max\frac{A_i \cap B_i}{A_i \cup B_i}, \tag{11}$$

where i denote relation type. Finally,

$$Sim(a,b) = \begin{cases} 1,\ if\ S_{synsets}(a,b) > 0 \\ max(S_{neighborhoods}(a,b), S_{descriptions}(a,b))\ if\ S_{synsets}(a,b) = 0 \end{cases} \tag{12}$$

where $S_{descriptions}$ means the matching of term description sets. $S_{descriptions}$ and $S_{synsets}$ are calculated according equation 11.

Bath et al. adapt Jaro-Winkler (JW) metric to integrate WordNet or EuroWordNet in processing ontology matching [1]. Name similarity (NS) of two names N_1 and N_2

of two classes A and B (each name is a set of tokens, $N = \{n_i\}$) is defined as

$$NS^{'}(N_1, N_2) = \frac{\sum_{n_1 \in N_1} MJW(n_1, N_2^{'}) + MJW(n_2, N_1^{'})}{|N_1| + |N_2|} \tag{13}$$

where $MJW(n_i, N) = max_{n_j \in N} JW(n_i, n_j)$, $N_i^{'} = N_i \cup \{n_k | \exists n_j \in N_i \bigcap n_k \in synset(n_j)\}$, $synset(n_j)$ is the set of synonyms of term n_j, $NS^{'}(A,B) = NS^{'}(N_1, N_2)$.

3.4 Applying WordNet Based Semantic Similarity Methods in Ontology Matching

Before applying the semantic similarity method in ontology matching, linguistic normalization is processed. Linguistic technologies transform each term to a standard form that can be easily recognized.

- Tokenisation consists of segmenting strings into sequences of tokens by a tokeniser which recognizes punctuation, cases, blank characters, digits, etc [6]. For example, $travel - agent$ becomes $< travel\ agent >$.
- Stemming is trying to remove certain surface marking words to root form. For example, words like *fishes* original form is *fish*.
- Stop-word [2] means that some words frequently appear in the text with lack of indexing consequence. Indexing is the process of associating one or more keywords with each document in information retrieval. For example, words like *the, this* and *of* in English, they appear often in sentences but have no value in indexing.
- Multiple part-of-speech. Each part-of-speech explains not what the word is, but how the word is used. In fact, the same word can be more than one part-of-speech (for instance, *backpacking* is both a noun and a verb in WordNet). When we compare the concept names which are made of single noun or noun phrase in the ontology, for these words it will be checked if they are nouns and if the answer is yes, we treat them as noun and disregard as verb [24].

WordNet based semantic similarity methods (see section 3.1, 3.2 and 3.3) can be used in two ways.

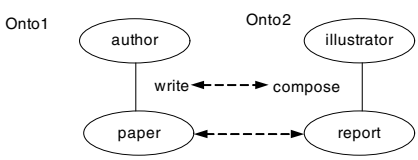


Fig. 2 Two simple ontologies.

- WordNet based semantic similarity methods can be applied to calculate entities similarities in two ontologies. For example, Figure 2 shows two simple ontolo-

gies Onto1 and Onto2. Property *write* in Onto1 and *compose* in Onto2 are synonyms in WordNet, we treat the labels of these two properties as equal even their string similarities are different. Since *paper* in Onto1 is the synonym of *report* in Onto2, they are treated as similar also.

There are two senses for the entry noun *author* hypernym relation in WordNet (version 2.1):

Sense 1

writer, author – (writes (books or stories or articles or the like) professionally (for money))

$\Longrightarrow$ *communicator – (a person who communicates with others)*

$\Longrightarrow$ *person, individual, someone, somebody, mortal, soul – (a human being; "there was too much for one person to do")*

$\Longrightarrow$ *organism, being – (a living thing that has (or can develop) the ability to act or function independently)*

...

Sense 2

generator, source, author – (someone who originates or causes or initiates something; "he was the generator of several complaints")

$\Longrightarrow$ *maker, shaper – (a person who makes things)*

$\Longrightarrow$ *creator – (a person who grows or makes or invents things)*

$\Longrightarrow$ *person, individual, someone, somebody, mortal, soul – (a human being; "there was too much for one person to do")*

...

There is one sense for the entry noun *illustrator* hypernym relation in WordNet (version 2.1):

illustrator – (an artist who makes illustrations (for books or magazines or advertisements etc.))

$\Longrightarrow$ *artist, creative person – (a person whose creative work shows sensitivity and imagination)*

$\Longrightarrow$ *creator – (a person who grows or makes or invents things)*

$\Longrightarrow$ *person, individual, someone, somebody, mortal, soul – (a human being; "there was too much for one person to do")*

...

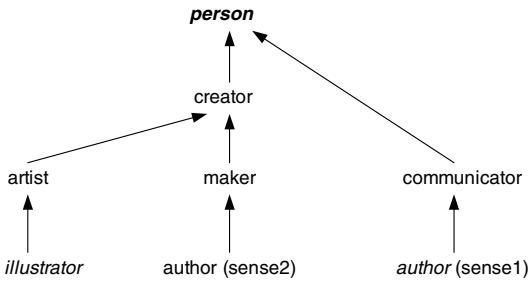


Fig. 3 The fragment of noun senses with *author* and *illustrator* in WordNet taxonomy.

Figure 3 presents the fragment of nouns with *author* and *illustrator* in WordNet taxonomy. If *author*is used in Onto1 and *illustrator* is used in Onto2 (see Figure 2), they have the common superconcept (hypernym) *person* in WordNet (see Figure 3), and we can apply WordNet based semantic similarity methods (see section 3.1, 3.2 and 3.3) to get similarity between *illustrator* and *author*.

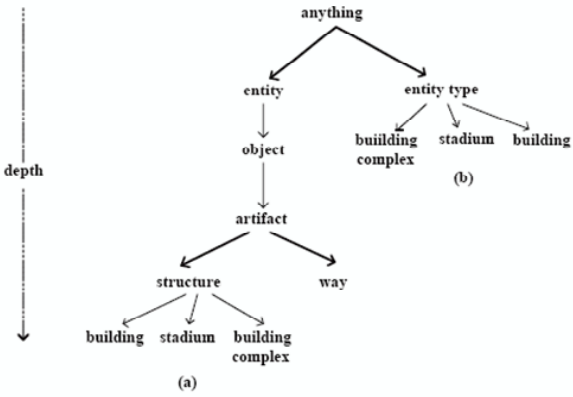


Fig. 4 Connecting independent ontologies: (a) partial WordNet ontology and (b) partial SDTS ontology. Source: [21]

- Rodriguez method [21] and X-Similarity [4] are independent from WordNet. They can be applied in ontology matching directly as structure similarity method if two independent ontologies have a common superconcept. For example, Figure 4 (see source [21]) shows two independent ontologies, *anything* is their common superconcept. Based on string similarity results, the structure similarity (e.g., similarity between $building^w$ in WordNet and $building^s$ in SDTS) can be calculated through Rodriguez method [21] and X-Similarity [4].

3.5 Evaluation of Semantic Similarity Methods

WordNet-Similarity [26] has implemented several WordNet-based similarity measures, such as Leacock-Chodorow [10], Jiang-Conrath [9], Resnik [18], Lin [13], Hirst-St-Onge [8], Wu-Palmer [27], Banerjee-Pedersen [15], and Patwardhan [15] in a Perl package.

Petrakis et al. [4] implement a "Semantic Similarity System" and evaluate several semantic similarity measures: Rada [17], Wu-Palmer [27], Li [12], Leacock-Chodorow [10], Richardson [20], Resnik [19], Lin [13], Lord [16], Jiang-Conrath [9], X-Similarity [4], Rodriguez [21]. Their evaluation in the same ontology is based on Miller and Charles [14] with the human relevance results. The higher the correlation of a method, the better the method is (i.e., the closer it is to the results of human

judgement). They also evaluate Rodriguez [21] and X-Similarity [4] methods in different ontologies (ontology matching).

SimPack [23] implements methods such as Jiang-Conrath [9], Lin [13], Resnik [19]. These methods have been evaluate by Budanitsky and Hirst [3].

Table 1 compares different WordNet-based similarity measures in WordNet-Similarity, Semantic Similarity System and SimPack:

Table 1 Implemented WordNet-based similarity measures in WordNet-Similarity, Semantic Similarity System and SimPack

WordNet-Similarity	Semantic Similarity System	SimPack
Leacock-Chodorow [10]	Leacock-Chodorow [10]	
Jiang-Conrath [9]	Jiang-Conrath [9]	Jiang-Conrath [9]
Resnik [18]	Resnik [18]	Resnik [18]
Lin [13]	Lin [13]	Lin [13]
Hirst-St-Onge [8]		
Wu-Palmer [27]	Wu-Palmer [27]	
Banerjee-Pedersen [15]		
Patwardhan [15]		
	Rada [17]	
	Li [12]	
	Richardson [20]	
	Lord [16]	
	X-Similarity [4]	
	Rodriguez [21]	

4 Conclusions

In this paper, we present different WordNet-based semantic similarity measures from edge-based methods to information-based statistic methods and their hybrid methods. We also discuss how to apply them in the ontology matching. Finally, we show several tools that implemented the semantic similarity measures and their evaluation.

Acknowledgements Part of this work was financed by the Hamrin Foundation (Hamrin Stiftelsen), project Media Information Logistics. Special thanks to Lars Ahrenberg for valuable comments and suggestions on this paper.

References

1. Bach, T.L., Dieng-Kuntz, R., Gandon, F.: On ontology matching problems (for building a corporate semantic web in a multi-communities organization). In: Proc. 6th International

Conference on Enterprise Information Systems (ICEIS), pp. 236–243. Porto (PT) (2004)
2. Belew, R.K.: Finding Out About: A Cognitive Perspective on Search Engine Technology and the WWW. Cambridge University Press (2001)
3. Budanitsky, A., Hirst, G.: Evaluating wordnet-based measures of lexical semantic relatedness. Comput. Linguist. **32**(1), 13–47 (2006)
4. Euripides G.M. Petrakis Giannis Varelas, A.H.P.R.: Design and evaluation of semantic similarity measures for concepts stemming from the same or different ontologies. In: In 4th Workshop on Multimedia Semantics (WMS'06), pp. 44–52 (2006)
5. EuroWordNet: http://www.illc.uva.nl/eurowordnet
6. Euzenat, J., Shvaiko, P.: Ontology matching. Springer-Verlag (2007)
7. Ferrer-i-Cancho, R.: The structure of syntactic dependency networks: insights from recent advances in network theory. In: L. V., A. G. (eds.) Problems of quantitative linguistics, pp. 60–75 (2005)
8. Hirst, G., St-Onge, D.: Lexical chains as representation of context for the detection and correction malapropisms (1997)
9. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy (1997)
10. Leacock, C., Chodorow, M.: Combining local context and wordnet similarity for word sense identification. An Electronic Lexical Database pp. 265–283 (1998)
11. Leacock, C., Miller, G.A., Chodorow, M.: Using corpus statistics and wordnet relations for sense identification. Comput. Linguist. **24**(1), 147–165 (1998)
12. Li Y.; Bandar, Z.M.D.: An approach for measuring semantic similarity between words using multiple information sources. Transactions on Knowledge and Data Engineering **15**(4), 871–882 (July-Aug. 2003). DOI 10.1109/TKDE.2003.1209005
13. Lin, D.: An information-theoretic definition of similarity. In: Proc. 15th International Conf. on Machine Learning, pp. 296–304. Morgan Kaufmann, San Francisco, CA (1998)
14. Miller George A., C.W.G.: Contextual correlates of semantic similarity. Language and Cognitive Processes **6**, 1–28 (1991)
15. Patwardhan, S., Banerjee, S., Pedersen, T.: Using Measures of Semantic Relatedness for Word Sense Disambiguation. In: Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics, pp. 241–257. Mexico City, Mexico (2003)
16. P.W.Lord R.D. Stevens, A.B., C.A.Goble: Investigating semantic similarity measures across the gene ontology: (2002)
17. Rada, R., Mili, H., Bicknell, E., Blettner, M.: Development and application of a metric on semantic nets. Systems, Man and Cybernetics, IEEE Transactions on **19**(1), 17–30 (1989)
18. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: IJCAI, pp. 448–453 (1995)
19. Resnik, P.: Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. Journal of Artificial Intelligence Research **11**, 95–130 (1999)
20. Richardson, R., Smeaton, A.F., Murphy, J.: Using WordNet as a knowledge base for measuring semantic similarity between words. Tech. Rep. CA-1294, Dublin, Ireland (1994)
21. Rodriguez, M., Egenhofer, M.: Determining semantic similarity among entity classes from different ontologies. IEEE Transactions on Knowledge and Data Engineering **15**(2), 442–456 (2003)
22. SemanticWeb: http://www.semanticweb.org/
23. SimPack: http://www.ifi.unizh.ch/ddis/simpack.html
24. Su, X.: Semantic enrichment for ontology mapping. Ph.D. thesis, Dept. of Computer and Information Science, Norwegian University of Science and Technology (2004)
25. WordNet: http://wordnet.princeton.edu
26. WordNet-Similarity: http://www.d.umn.edu/~tpederse/similarity.html
27. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: 32nd. Annual Meeting of the Association for Computational Linguistics, pp. 133 –138. New Mexico State University, Las Cruces, New Mexico (1994)
28. Yi Li Jie Tang, D.Z.J.L.: Toward strategy selection for ontology alignment. In: Proceedings of the 4th European Semantic Web Conference 2007 (ESWC2007) (2007)

Using Competitive Learning between Symbolic Rules as a Knowledge Learning Method

F. Hadzic[1] and T.S. Dillon[2]

Abstract We present a new knowledge learning method suitable for extracting symbolic rules from domains characterized by continuous domains. It uses the idea of competitive learning, symbolic rule reasoning and it integrates a statistical measure for relevance analysis during the learning process. The knowledge is in form of standard production rules which are available at any time during the learning process. The competition occurs among the rules for capturing a presented instance and the rules can undergo processes of merging, splitting, simplifying and deleting. Reasoning occurs at both higher level of abstraction and lower level of detail. The method is evaluated on publicly available real world datasets.

1 Introduction

Within the AI community there are two views on how a machine should engage in the knowledge learning task. Some believe that the human learning should be mimicked in the way that it is initially learned at the lower neuronal level (connectionism) while others believe it is the higher level conceptual reasoning with symbolic rules (symbolism) that will bring machine learning closer to human learning [1]. Anderson [2] believes that both neural aspects and cognitive rules exist and that it is unlikely to have an accurate psychological theory without the formulation of explicit rules that enable proper representation of the acquired generalizations. Chandasekaran et al. [3] stated that connectionism and symbolism

[1] F. Hadzic

Digital Ecosystems and Business Intelligence Institute, GPO Box U1987 Perth, Australia email: fedja.hadzic@postgrad.curtin.edu.au

[2] Prof. T.S.Dillon

Digital Ecosystems and Business Intelligence Institute, GPO Box U1987 Perth, Australia email: tharam.dillon@cbs.curtin.edu.au

Please use the following format when citing this chapter:

Hadzic, F. and Dillon, T.S., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 351360.

both agree on the view of intelligence as information processing of representations, but disagree on the storage and processing mechanism of those representations. These observations, together with the advantages and disadvantages of symbolic and sub-symbolic systems, justifies the large amount of research that has gone into the extraction of symbolic knowledge from neural networks.

In [4] we presented a variation of the Self-Organizing Map (SOM) [5], CSOM, that applies a different learning mechanism useful for situations where the aim is to extract rules from a data set characterized by continuous input features. Attribute constraints for continuous attributes are contained on the network links themselves, thereby making the network itself more symbolic. A rule optimizing method was integrated into CSOM and in [6] it was adjusted so that it can be applied to rule sets obtained using other knowledge learning methods. Its potential of optimizing rules set through symbolic reasoning has motivated us to investigate whether the method itself can be extended so that it becomes a stand-alone knowledge learning method.

The aim of the work presented in this paper is to present a new knowledge learning method that combines symbolic and sub-symbolic reasoning, but is capable of presenting its acquired knowledge at any time with symbolic rules. It is motivated partly by the competitive learning and progressive parameter adjustment that occurs in the SOM [5]. The main difference is that the competition occurs among the rules for capturing the instances rather than neurons and hence symbolic rules are available at any time during learning. The system is able to move from the lower level where weight interactions are used for reasoning to the higher level where symbolic rules are represented and reasoned with. The proposed method can be viewed as an intersection of competitive learning, symbolic reasoning and statistics.

2 Method Description

The dataset describing the domain at hand is split into an unsupervised (class labels removed) and supervised (class labels present) dataset. Initially the rule set is empty and the unsupervised file is used according to which the initial rule set is obtained. The main part according to which the learning is done is the structure used to represent the currently learned rules which are in form of antecedent-consequent pairs. Initially, the rule set is empty and as the instances from a file are read in rules are set up whose antecedents are equal to the attribute values occurring in the samples. When a set of instances is fed on top of the rule set, it can be said that competition occurs between the rules for capturing the presented instances. The rule that most closely matches the instance in terms of attribute constraints captures that instance and its attribute constraints are adjusted if necessary. During this process similar rules may be merged while contradicting

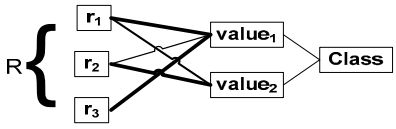
rules may be split. The merging of rules is highly dependent on the threshold chosen according to which two rules are considered similar in terms of attribute constraints. Two learning parameters are used which are progressively adjusted during the learning. One parameter is for allowing an instance to be captured by the rule (*IR*) and the second is for allowing two rules to merge together (*MR*). Setting both of these parameters to very low values initially will result in a large rule set. As learning proceeds, both parameters are increased. By increasing the *IR* the set of rules will capture more instances thereby increasing the coverage rate of the rules and decreasing the number of new rules that need to be formed. By increasing the *MR* similar rules are more likely to merge into one making the rule set more general and reducing the number of rules. Splitting of rules occurs when a misclassification occurs. Similarly the instance information about attribute and class relationships is collected which allows for the reasoning to occur at the lower level (sub-symbolic). At this level the relevance of attributes for a particular rule is determined using a statistical measure. The process as a whole is repeated for a chosen number of iterations.

2.1 Representative Structure

Let m denote the number of input attributes (denoted as x_i) in a dataset D from which the knowledge is to be learned. An instance or example e from D will be referred to as an input vector and denoted by $IV_e = (x_1, x_2, \ldots, x_m)$. Further let x_t denote the class value associated with example e. Let R denote the set of rules in the structure. A rule will be generally denoted as r_p, where $p = (1, \ldots, |R|)$. The attribute constraints of each rule (i.e. the antecedent part of the rule) are contained in the weight vector of that rule (denoted as $WV(r_p)$). An attribute constraint at position i in $WV(r_p)$ is denoted as a_i. Even though some rules will not contain the total set of attributes in their WV the ordering is kept so that items at the same index positions in the IV and the WV correspond to the values or constraints of same attributes in the dataset (i.e. $x_i = a_i$). Hence, to classify an instance e we match the IV_e against the weight vectors of the available rules using a modification of the Euclidean distance (*ED*) measure as the basis for comparison. This process is explained in detail later in the paper. Each rule r_p has a target vector associated with it denoted as $TV(r_p)$ which contains the links to class values that have occurred in the instances that are covered by that particular rule. Let an item at position t in $TV(r_p)$ be denoted as tv_t and the weight associated with it as $w(tv_t)$. The implying class value of a rule r_p becomes the one which has the highest weight associated with it. In other words if the implying class corresponds to the item in $TV(r_p)$ at position x then $\max(w(tv_t)) = w(tv_x) \; \forall \; t \mid t = (1, \ldots, |TV(r_p)|)$. This class value has most frequently occurred in the instances that were captured by the rule. An example of the structure representing the rule set with target vector information is shown in Figure 1 (thicker lines correspond to higher weights on links). The implying class of rules r_1 and r_3 would be class value1 while for rule r_2

it is class value2. Even though it is not shown in Figure 1, the attribute constraints associated with each rule are stored in the *weight vector* of that rule.

Figure 1: Example structure representing the rule set and related information.



2.1.1 Storing Low-Level Information

A constraint for a continuous attribute is given in terms of a lower range (*lr*) and an upper range (*ur*) indicating the set of allowed attribute values. For each $a_i \in WV(r_p)$, where $i = (1, \ldots, m)$ the occurring attribute values in the instances that were captured by that particular rule are stored in a value list (denoted as $VL(a_i)$). The items in $VL(a_i)$ are referred to as value objects and an item at position r in $VL(a_i)$ is denoted as v_r. Each $v_r \in VL(a_i)$, where r = (1,…, $|VL(a_i)|$) has a target vector associated with it, denoted as $TV(v_r)$ which contains the links to class values that have occurred together with that particular value in the instance captured by the rule.

Since for continuous attributes there could be many occurring values, close values are merged into one value object when the difference between the values is less than a chosen merge value threshold. Hence a value object v_r may either contain a singular value, denoted as v_{rVAL} or a lower limit and the upper limit on the range of values, denoted as v_{rLR} and v_{rUR}, respectively. If an attribute value at position i in IV_e is denoted as x_i and the target or class value as x_t then the update process can be summarized as follows. When a rule r_p captures an instance then a link to x_t is added to $TV(r_p)$ with weight set to 1, or the weight on the link is incremented if $TV(r_p)$ already contains a link to that particular x_t. For each $a_i \in WV(r_p)$ the $VL(a_k)$ is updated by either incrementing the weight of a v_r if $x_i = v_{rVAL}$ or $v_{rLR} <= x_i <= v_{rUR}$ *if* v_r ranges are set or otherwise adding a new v_r ($v_{rVAL} = x_i$) (i.e. inserting x_i value in position r in $VL(a_i)$) such that $v_{(r-1)VAL} < v_{rVAL} < v_{(r+1)VAL}$ or if the $v_{(r-1)}$ and $v_{(r+1)}$ are ranged value objects then $v_{(r-1)UR} < v_{rVAL} < v_{(r+1)LR}$. Furthermore, a link to x_t is added to $TV(v_r)$ with weight set to 1, or the weight on the link is incremented if $TV(v_r)$ already contains a link to that particular x_t. Hence the numerical values stored in *VL* of an attribute will be ordered so that a new value is always stored in an appropriate place and the merging can occur if necessary.

Figure 2 illustrates how this low level information is stored for a rule that consists of two continuous attribute a_1 and a_2, and points to two class values (i.e. Value1 and Value2). For example the attribute a_1 has the lower range and the upper range in between which the values v_1, v_2 and v_3 have occurred. The lower range of a_1 is equal to v_{1VAL} or the v_{1LR} if v_1 is a merged value object, while the upper range of a_1 is equal to v_{3VAL} or the v_{3UR} if v_3 is a merged value object.

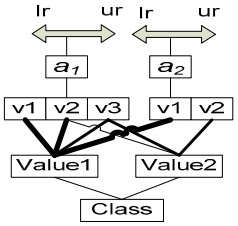


Figure 2: Storing low level instance information.

2.2 Measure for Capturing Instances and Rule Merging

To determine which particular rule should capture an instance and whether two rules should be merged we make use of a modified Euclidean distance (*ED*) measure. An instance *e* is always captured by the rule with the smallest *ED* between the rule's weight vector and the IV_e. The notation used for measuring *ED* between IV_e and a *WV* of a rule r_p is $ED(IV_e, WV(r_p))$. Similarly if we are measuring similarity among two rules r_1 and r_2 then the notation $ED(WV(r_1), WV(r_2))$ is used. The i-th term of $ED(IV, WV(r_p))$ is equal to the distance of the input attribute value x_i to the nearest range boundary of the i-th item in $WV(r_p)$.

Let the i-th term of the $ED(IV_e, WV(r_p))$ be denoted by $r_p term_i$. To determine which rule most closely matches the *IV* the following expression is used.

$$\arg\min\left(\sqrt{\sum_{i=1}^{n}\left(r_p term_i\right)^2}\right)$$

$$\forall\, p \mid p = \{1 \ldots |R|\}.$$

The *IR* needs to be set with respect to the number of attributes in the dataset. It corresponds to the maximum allowed sum of the range/value differences among the attributes of $WV(r_p)$ and IV_e so that the rule would capture the instance at hand. The instance is captured by the rule with the smallest *ED* between its weight vector and the IV_e. If no rule exactly matches IV_e *(i.e.* $ED(IV_e, WV(r_p)) \neq 0 \;\forall$ p = (1, …, $|R|$)) and no rule is close to the instance (i.e. $ED(IV_e, WV(r_p)) >$ *InstToRuleThr* $\forall$ p = (1, …, $|R|$)) then a new rule r_n will be created where all attribute values in its weight vector are set according to the instance attribute values (i.e. $a_i v = x_i \;\forall$ i = (1, …, *m*)).

When calculating the *ED* for the purpose of merging similar rules there are four possibilities that need to be accounted for with respect to the ranges being set in the rule attributes. Two rules r_1 and r_2 will be merged if the $ED(WV(r_1), WV(r_2)) <$ *MR*. For rule r_1 let $r_1 a_i$ denote the attribute occurring at the position *i* of $WV(r_1)$, let $r_1 a_i lr$ denote the lower range, $r_1 a_i ur$ the upper range, and $r_1 a_i v$ the initial value if the ranges of $r_1 a_i$ are not set. Similarly for rule r_2 let $r_2 a_i$ denote the attribute occurring at the position i of $WV(r_2)$, let $r_2 a_i lr$ denote the lower range, $r_2 a_i ur$ the

upper range, and 'r_2a_iv' the initial value if the ranges of r_2a_i are not set. The i-th term of the *ED(WV(r_1), WV(r_2))* calculation for continuous attributes is:

- case 1: both r_1a_i and r_2a_i ranges are not set

0 iff $r_1a_iv = r_2a_iv$

$r_1a_iv - r_2a_iv$ if $r_1a_iv > r_2a_iv$

$r_2a_iv - r_1a_iv$ if $r_1a_iv < r_2a_iv$

- case 2: r_1a_i ranges are set and r_2a_i ranges are not set

0 iff $r_2a_iv \geq r_1a_ilr$ and $r_2a_iv \leq r_1a_iur$

$r_1a_ilr - r_2a_iv$ if $r_2a_iv < r_1a_ilr$

$r_2a_iv - r_1a_iur$ if $r_2a_iv > r_1a_iur$

- case 3: r_1a_i ranges are not set and r_2a_i ranges are set

0 iff $r_1a_iv \geq r_2a_ilr$ and $r_1a_iv \leq r_2a_iur$

$r_2a_ilr - r_1a_iv$ if $r_1a_iv < r_1a_ilr$

$r_1a_iv - r_2a_iur$ if $r_1a_iv > r_2a_iur$

- case 4: both r_1a_i and r_2a_i ranges are set

0 iff $r_1a_ilr \geq r_2a_ilr$ and $r_1a_iur \leq r_2a_iur$

0 iff $r_2a_ilr \geq r_1a_ilr$ and $r_2a_iur \leq r_1a_iur$

$\min(r_1a_ilr - r_2a_ilr, r_1a_iur - r_2a_iur)$ iff $r_1a_ilr > r_2a_ilr$ and $r_1a_iur > r_2a_iur$

$\min(r_2a_ilr - r_1a_ilr, r_2a_iur - r_1a_iur$ iff $r_2a_ilr > r_1a_ilr$ and $r_2a_iur > r_1a_iur$

$(r_1a_ilr - r_2a_iur)$ iff $r_1a_ilr > r_2a_iur$

$(r_2a_ilr - r_1a_iur)$ iff $r_2a_ilr > r_1a_iur$

2.3 Reasoning Process

This section explains the reasoning processes that occurs with the information stored in the knowledge structure as explained in the previous sections.

2.3.1 Higher Level Reasoning

Once the implying classes are set for each of the rules, the dataset is fed on topf of the rules. When a rule captures an instance that has a different class value than the implication of the rule, a child rule is created in order to isolate the characteristic of the rule causing the misclassification. The attribute constraints of the parent and child rule are updated so that they are exclusive from one another. In other words, in future an instance will be either captured by the parent rule or the child rule, not both. After the whole dataset is read in, there could be many child rules created from a parent rule. If a child rule points to other target values with high confidence it becomes a new rule. This corresponds to the process of rule splitting, since the parent rule has been modified to exclude the child rule which is now a rule on its own. On the other hand if the child rule still mainly points to the implying class value of the parent rule it is merged back into the parent rule. An example of a rule for which a number of children were created due to the misclassifications is displayed in Figure 3. The reasoning explained would

merge 'Child3' back into the parent rule, while Child1 and Child2 would become new rules.

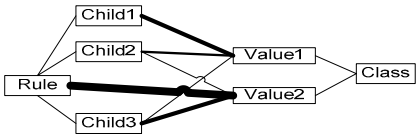


Figure 3: Example of rule splitting.

Rule Merging. After the whole file is read in the child rules that have the same implying class values are merged together if the ED between them is below the *MR*. Once all the child rules have been validated the merging can occur in the new rule set. Hence if two rules r_1 and r_2 have the same implying class value and the $ED(WV(r_1), WV(r_2)) < MR$ the rules will be merged together and the attribute constraints updated. Rather than creating a new rule at the implementation level the merged rule is one of the original rules (say r_1) with its weight vector updated accordingly, while the second rule is removed from the rule set. The update is done in such manner so that the attribute constraints in the $WV(r_2)$ are contained within the attribute constraints of $WV(r_1)$ *(i.e.* $ED(WV(r_1), WV(r_2)) = 0$). Hence, if the range of allowed values for an attribute in the $WV(r_2)$ fell outside the corresponding attribute value range in $WV(r_1)$ then that particular range is expanded to include the value range in $WV(r_2)$. More formally the process can be explained as follows.

The same notation as earlier will be used where the items occurring at position i in $WV(r_1)$ and $WV(r_2)$ will be denoted as r_1a_i and r_2a_i, respectively. If the ranges on the items are not set, then the same notation r_1a_i and r_2a_i also corresponds to the initial values of those attributes. Otherwise $_{LR}$ or $_{UR}$ is appended for lower range or upper range respectively (eg. r_sa_{iLR}). Depending on whether the range of the items in the weight vector is set, there are four cases which determine in which way the $WV(r_1)$(the new merged rule) is updated. It can be expressed as follows. In the logic expressed below the cases when no update needs to occur were excluded. These are the cases when the value(s) of r_2a_i was either equal to the value of r_1a_i or if ranges on r_2a_i are set they fell within the ranges of r_1a_i.

Case 1: r_1a_i and r_2a_i ranges are not set

```
If r1ai  > r2ai
  r1aiUR = r1ai
  r1aiLR = r2ai
If r1ai  < r2ai
  r1aiUR = r2ai
  r1aiLR = r1ai
```

Case 2: r_1a_i range is set and r_2a_i range is not set

```
If r2ai > r1aiUR
  r1aiUR = r2ai
If r2ai < r1aiLR
```

$r_1a_{iLR} = r_2a_i$

Case 3: r_1a_i range is not set and r_2a_i range is set

```
If r1ai > r2aiUR
 r1aiUR = r1ai
 r1aiLR = r2aiLR
If r1ai < r2aiLR
 r1aiUR = r2aiUR
 r1aiLR = r1ai
If r2aiLR <= r1ai <= r2aiUR
 r1aiUR = r2aiUR
 r1aiLR = r2aiLR
```

Case 4: both r_1a_i and r_2a_i ranges are set

```
If r2aiLR < r1aiLR
  r1aiLR = r2aiLR
If r2aiUR > r1aiUR
  r1aiUR = r2aiUR
```

2.3.2 Reasoning at the Lower Level

This section describes the process of reasoning with the instance information collected at the lower level of the structure as described in Section 2.1.1. Once the rules have undergone the process of splitting and merging, the relevance of rule attributes should be calculated as some attributes may have lost their relevance through merging of two or more rules. Other attributes may have become relevant as a more specific distinguishing factor of a new rule that resulted from splitting of an original rule. Hence this process happens after a number of iterations where reasoning at the higher level of the structure occurred. The Symmetrical Tau (**τ**) [7] feature selection criterion is used and its calculation is enabled with the instance information collected at the lower level of the structure (Section 2.1.1).

Simplification or rules using the τ criterion. The first step is to calculate the τ measure for each attribute a_i (where i = (1, …, *m*)) in the weight vector $WV(r_p)$ of a rule r_p. Once the τ measure has been calculated for each attribute a_i in $WV(r_p)$ all a_i are ranked according to decreasing τ value. A relevance cut-off is determined in the ranking and it occurs at an attribute if its τ value is less than half of the previous attribute's τ value in the ranking. At this point and below in the ranking all attributes are considered as irrelevant for that rule. On the other hand, if some of the attributes above the relevance cut-off point were previously excluded from $WV(r_p)$, they are now re-introduced since their τ value indicates their relevance for the rule at hand. Once the attribute relevance has been determined for all rules the whole process of rule optimization continues with the main difference being that not all terms in the *ED* formula will be calculated since some attribute constraints do not form the necessary part of the rule any more. When calculating the *ED* between a rule r_p and the input vector IV_e, the i^{th} term of the *ED* formula will be excluded if the attribute a_i is irrelevant. Similarly when calculating the *ED* between two rules r_1 and r_2 the i^{th} term of the *ED* formula will be excluded if the

attribute a_i is irrelevant for both rules. If a_i is irrelevant for one of the rules but not for both, then the rules will not be considered similar to be merged.

3 Method Evaluation

This section describes some experiments performed on a number of real world datasets obtained from the 'uci' machine learning repository [8]. The training dataset was made up of about 70% of randomly chosen instances from the original dataset while the rest was used for testing. Due to space limitations, a detailed comparison with other knowledge learning methods is not provided, but the results are comparable to those obtained by other inductive learners.

Table1: Learned rules from the 'Iris' dataset

Rule 1: 0 < sepal-length < 0.417 AND 0.125 < sepal-width < 1.0 AND 0 < petal-length < 0.153 AND 0 < PW < 0.208 → Iris-setosa
Rule 2: 0.644 < petal-length < 1.0 AND 0.542 < petal-width < 1.0 → Iris-virginica
Rule 3: 0.361 < sepal-length < 0.472 AND 0.417 < sepal-width < 0.583 AND 0.593 < petal-length < 0.644 AND 0.583 < petal-width < 0.708 → Iris-versicolor
Rule 4: 0 < sepal-width < 0.542 AND 0.339 < petal-length < 0.695 AND 0.375 < petal-width < 0.667 → Iris-versicolor

Table 2: Learned rules from the 'Wine' dataset

Rule 1: 0.0 < Alcohol < 0.74 AND 0.03 < Malic_acid < 1.0 AND 0.18 < Ash < 1.0 AND 0.09 < Magnesium < 0.53 AND 0.04 < Total_phenols < 0.88 AND 0.14 < Flavanoids < 1.0 AND 0.05 < Color_intensity < 0.4 AND 0.2 < Hue < 1.0 AND 0.2 < OD280/OD315_diluted_wines < 0.89 AND 0.0 < Proline < 0.43 → Two
Rule 2: 0.48 < Alcohol < 1.0 AND 0.12 < Malic_acid < 0.65 AND 0.36 < Ash < 0.99 AND 0.03 < Alcalinity_of_ash < 0.74 AND 0.21 < Magnesium < 0.67 AND 0.42 < Total_phenols < 1.0 AND 0.39 < Flavanoids < 0.76 AND 0.08 < Nonflavanoid_phenols < 0.7 AND 0.26 < Proanthocyanins < 0.8 AND 0.19 < Color_intensity < 0.65 AND 0.28 < Hue < 0.65 AND 0.45 < OD280/OD315_diluted_wines < 1.0 AND 0.29 < Proline < 1.0 → One
Rule 3: 0.31 < Alcohol < 0.87 AND 0.1 < Malic_acid < 0.97 AND 0.4 < Ash < 0.8 AND 0.36 < Alcalinity_of_ash < 0.85 AND 0.11 < Magnesium < 0.58 AND 0.0 < Total_phenols < 0.63 AND 0.0 < Flavanoids < 0.26 AND 0.08 < Nonflavanoid_phenols < 0.94 AND 0.04 < Proanthocyanins < 0.72 AND 0.22 < Color_intensity < 1.0 AND 0.0 < Hue < 0.39 0.0 < OD280/OD315_diluted_wines < 0.44 AND 0.1 < Proline < 0.43 → Three
Rule 4: 0.1 < Alcohol < 0.52 AND 0.0 < Malic_acid < 0.59 AND 0.0 < Ash < 0.74 AND 0.16 < Total_phenols < 0.8 AND 0.05 < Flavanoids < 0.59 AND 0.02 < Nonflavanoid_phenols < 0.94 AND 0.0 < Color_intensity < 0.38 AND 0.17 < Hue < 0.79 0.12 < OD280/OD315_diluted_wines < 0.82 AND 0.03 < Proline < 0.5 → Two

The learning parameters for *Iris* dataset were set as follows: *iteration#* = 100, *MR* was progressively increased from 0.02 to 0.1 whereas *IR* from 0.00001 to 0.05. The merge value threshold used for merging the value objects of an attribute (see Section 2.1.1) was set to 0.02. The learned rules are displayed in Table 1. Overall the rule set had 93.81% classification accuracy and 96% prediction accuracy. For the *Wine* dataset the following learning parameters were used:

iteration# = 60, *MR* was progressively increased from 0.05 to 0.5 whereas *IR* from 0.01 to 0.05. The merge value threshold was set to 0.04. A total of 4 rules were obtained which are displayed in Table 2 with the classification accuracy of 98.4% and the predictive accuracy of 98.1%.

4 Conclusion

This paper presented a new knowledge learning method for continuous domains that combines the idea of competitive learning, symbolic rule reasoning and statistics. The main difference with the traditional competitive learning as used in the Self-Organizing Map is that the competition occurs among symbolic rules rather than network units. Hence one useful property of the method is that symbolic rules are available at any time during the learning process. The integration of the statistical feature selection criterion has proven useful for attribute relevance analysis during learning and simplification of the learned rules. Evaluation of the method on real world dataset has demonstrated the effectiveness of the method for extraction of optimal symbolic rules. As part of the future work this method will be evaluated on more complex real world datasets and compared more closely with some of existing rule based systems.

References

1. Sestito, S. and Dillon, S.T.: *Automated Knowledge Acquisition*. Prentice Hall of Australia Pty Ltd, Sydney, (1994).
2. Anderson, J.R.: *The Architecture of Cognition*, Harvard University Press, Cambridge, Massachussetts, (1983).
3. Chandrasekaran, B., Goel, A. and Allemang, D.: Connectionism and information processing abstractions. In *AI Magazine*, vol.9, no. 4, Winter, pp. 25-34, American Association for Artificial Intelligence, (1988).
4. Hadzic, F. & Dillon, T.S.: CSOM: Self Organizing Map for Continuous Data, In *Proceedings of the 3rd International IEEE Conference on Industrial Informatics (INDIN'05)*, 10-12 August, Perth, (2005).
5. Kohonen, T.: The Self-Organizing Map. In *Proceedings of the IEEE*, Vol. 78, no 9, pp. 1464-1480, September, (1990).
6. Hadzic, F. and Dillon, T.S.: Rule Optimizing Technique Motivated by Human Concept Formation, In *Proceedings of the International Conference on Bio-Inspired Systems and Signal Processing (BIOSIGNALS 2008)*, January 28-31, Funchal, Madeira, Portugal, (2008).
7. Zhou, X.-J.M. and Dillon, T.S.: A statistical-heuristic feature selection criterion for decision tree induction. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8), 8, 1991, pp. 834-841, (1991).
8. Blake, C., Keogh, E. and Merz, C.J.: UCI Repository of Machine Learning Databases. Irvine, CA: University of California, Department of Information and Computer Science (1998) [http://www.ics.uci.edu/~mlearn/MLRepository.html].

Knowledge Conceptualization and Software Agent based Approach for OWL Modeling Issues

S. Zhao[1], P. Wongthongtham[2], E. Chang[3], and T. Dillon[4]

Abstract In this paper, we address the issues of using OWL to model the knowledge captured in relational databases. Some types of knowledge in databases cannot be modeled directly using OWL constructs. Two alternative approaches are proposed with examples of two types of knowledge. Firstly the data value range constraint and secondly the calculation knowledge representation. The first approach to the problem is to conceptualize the data range as a new class and the second solution to the problem is proposed, based on utilizing software agent technology. Examples with OWL code and implementation code are given to demonstrate the problems and solutions.

1 Introduction

With the increasing trend of collaborations amongst organizations and business needs for sharing and publishing their products information, information and knowledge held in vast number of databases are demanded to be shared and integrated without organizational and application boundaries. However, databases are enterprise and application dependant in that their design and development are subjected to a particular business problem domain of an organization. This has prevented the databases from being shared and integrated in an open environment.

Ontology-based technologies provide a feasible approach to this problem. Ontology-based technologies promote knowledge sharing and integration by formally and explicitly defining the meanings and associations of information and data. An ontology is defined as "*a formal, explicit specification of shared conceptualization*" [1-3]. Ontologies allow specially designed software agents to automatically process and integrate information from distributed sources. Many approaches have been proposed to transform the knowledge embedded in databases, particularly in relational databases, into ontologies [4-9]. The transformation process involves database reverse engineer-

1 2 3 4.Shuxin Zhao, Dr. Pornpit Wongthongtham, Prof. Elizabeth Chang, Prof. Tharam Dillon

Digital Ecosystems & Business Intelligence Institute, Curtin University of Technology, Australia.
email: {s.zhao, p.wonthongtham, e.chang, tharam.dillon}@curtin.edu.au

Please use the following format when citing this chapter:

Zhao, S., Wongthongtham, P., Chang, E. and Dillon, T., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 361370.

ing for acquiring the implicit knowledge from databases and involves mapping the extracted knowledge onto an ontology language. OWL [10], particularly OWL DL, as the WWW consortium recommendation for Semantic Web, has gained the popularity as the target ontology language. Hereafter, we refer OWL in this paper to OWL DL, as it's the most practical one among the three sub-languages for Semantic Web.

However, there is a critical issue of using OWL, to fully and accurately represent the knowledge captured in relational databases. Although there are many similarities between an ontology and a conceptual data model of a database, such as UML or EER model, there are many practical issues when mapping the knowledge captured in a conceptual model onto an OWL ontology. For example, there are three common types of relationships between concepts we model in an UML model, namely, generalization/specialization, aggregation and composition and association. While generalization/specialization can be modeled straightforwardly using OWL hierarchical mechanism i.e. *Class* and *Subclass, Property* and *Subproperty,* the aggregation/composition relationship cannot be represented directly using OWL elements. There are also other types of knowledge captured by relational databases that we found hard to represent using OWL constructs such as the value range restrictions on an attribute, and the functional dependency among several attributes of one or more tables which captures some sort of relationships between attributes rather than concepts.

In this paper, we present two alternative solutions to tackle this OWL modeling issue, namely, conceptualization approach and software agent based approach. Two specific examples are used to demonstrate each of the approaches respectively: firstly the problems of modeling the data value range constraint; secondly, the problem of modeling mathematic calculation knowledge, whose operands are derived from attributes of one or more concepts, which represents relationships between these attributes. Our motivation is to reveal some ideas of extending the expressiveness of OWL in the mean time to retain computational completeness of the ontology model, thus to make OWL more adaptable to various domain knowledge representations.

The rest of this paper is organized as follows: Section 2 reviews related work on these issues; Section 3 describes the problems in details with examples; followed by Section 4 demonstrating the solutions to the problems with code example; last in Section 5, we conclude the paper and indicate future work.

2 Related Work

OWL provides powerful mechanisms to enhance reasoning about the *Classes* and relationships amongst *Classes* but not for representing and reasoning relationships between *Properties*. The OWL modeling issue originates from its design pursuit of the trade-off between expressiveness and scalability of a language. Most important kinds of knowledge are supported in OWL, particularly for sub-assumption and classification, while the computational completeness and decidability must also be retained [11, 12]. As a consequence, the OWL is designed to be maximum expressive without being undecidable. This has resulted in the expressiveness limitations amongst other OWL weakness which are identified in [13]. One of W3C's solutions to this problem is to introduce *Rules* (*RIF*) [14] which aims to provide greater expressiveness in con-

junction with RDF/OWL, typically, to provide a richer language for representing dependencies between *Properties* rather than *Classes*. *RIF Core Design* working draft has been released in Oct 2007. One plausible drawback to this Rule-based solution is that the knowledge needs to be encoded by more than one or two languages in order to represent the full domain.

Besides of the above, there is not much work that has been reported on addressing the issues of the knowledge representation with OWL. Stojanovic et al. [5] mentioned that some database related dynamic knowledge embedded in SQL stored procedures, triggers and built-in functions cannot be mapped to RDF.

3 Problem Description with Examples

In this section, we describe the two specific types of knowledge that cannot be modeled directly using OWL constructs in order to demonstrate the idea of tackling the above mentioned modeling issues. Solutions to the problems are given in the following section.

3.1 Data Value Range Modeling Problem in OWL

The first type of knowledge that we mentioned in the introduction section that cannot be modeled directly using constructs, which are specified in OWL DL, is the constraint on data value range. Data value range constraint is very common to various domains. For example, a company recruitment statement contains a minimum age and a maximum age requirement and a bank product requests a minimum and a maximum amount of deposit over a period such as monthly. This refers to data value range in database development. This kind of data constraint can be obtained from database schema, application source code through validation and SQL queries. It, however, cannot be directly represented using any constructs specified in OWL DL. One example of the recruitment requirement for the employee's age constraint in a company, named ABC, can be expressed as the formula:

```
ABCEmployee (18 < age < 65)
```

In OWL DL, if we define a Class namely *Employee*, with a *DatatypeProperty* namely *age* shown as in the OWL definition below:

```
<owl:Class rdf:ID="Employee"/>
<owl:DataTypeProperty rdfID="age">
      <rdfs:domain rdfresource="#Employee" />
      <rdfs:range rdfresource="&xsd;integer" />
<owl:DatatypeProperty>
```

We may further add constraints such as the cardinality on the *age* property, but no any other elements defined in OWL for property restrictions, such as *allValueFrom* and the set operator like *unionOf* and *intersectionOf*, can be used to model the simple

value range constraint. We therefore need other means to represent this kind of knowledge in OWL ontologies.

3.2 Calculation Knowledge Representation Problem in OWL

The second type of knowledge cannot be modeled directly using OWL constructs is the general calculation knowledge. An arithmetic calculation consists of operands and arithmetic operators such as addition, subtraction, multiplication and division. Operands in a calculation are often derived from columns of tables in a database or from properties of Classes in an ontology. The result of a calculation, in the mean time, is assigned to a column or a property. This represents associations amongst properties rather than classes. It may also represent the dynamic knowledge which is generated at run time in a given application. This type of knowledge is usually defined in SQL queries such as stored procedures or application source code when validating new data entry to ensure data consistency. One example of this type of knowledge is the calculation of total cost including GST tax of a purchase. The cost is calculated based on three properties: the *"quantity"* of the product in the purchase, the *"price"* of the product excluding GST tax and current *"GST tax rate"*. It can be expressed as the following formulas:

```
SubTotal = itemQuantity * singleUnitPrice
Tax = SubTotal * GSTRate
TotalCost = SubTotal + Tax
```

In OWL, there is no constructs defined for modeling this type of associations among properties from one or more Classes.

4 Approach

For the modeling problems stated in the previous section, we propose two alternative approaches to tackle the issues. They are described and demonstrated with sample code in this section.

4.1 Conceptualization of Data Value Range Constraint in OWL

As OWL does not provide any constructs for restricting value range on *DatatypePropertie*, we cannot represent this constraint directly in the way that we specify it in a programming language or in a database management system. However, we can model the value range constraint by conceptualizing it into a new Class. The conceptualization actually explicitly reflects the semantics of the data restriction because the general concept *Age* of human being is different from the concept *minimum age* and *maximum age* in a company recruitment requirement. We demonstrate the solution to the first problem defined in section 3.1 as in List 1.

In the OWL ontology List 1, the constraint on employee's age is conceptualized as a new Class *"EmploymentAge"*. It has two *DatatypeProperties*: *"minAge"* and *"maxAge"*. There is one individual created for ABC company recruitment requirement called *"ABCEmploymentAge"* whose *"minAge"* is 18 and *"maxAge"* is 65. The property *"age"* of the Class *"Employee"* can therefore be defined as an *ObjectProperty* whose range is of the class *"EmployeeAge"*. If there are individuals of ABC company employee, their age must be between 18 and 65. One key point to this solution is that this conceptualization must be transformed or mapped in implementation. Likewise, other types of knowledge can also be conceptualized in this way.

```
<owl:Class rdf:ID="EmploymentAge"/>

 <owl:DatatypeProperty rdf:ID="maxAge">
    <rdfs:domain rdf:resource="#EmploymentAge"/>
    <rdfs:range rdf:resource="&xsd#int"/>
  </owl:DatatypeProperty>

 <owl:DatatypeProperty  rdf:ID="minAge">
    <rdfs:range rdf:resource="&xsd#XMLSchema#int"/>
    <rdfs:domain rdf:resource="#EmploymentAge"/>
  </owl:DatatypeProperty>

  <EmploymentAge rdf:ID="ABCEmploymentAge">
     <maxAge rdf:datatype="&xsd#int">18</maxAge>
     <minAge rdf:datatype="&xsd#int">65</minAge>
  </EmploymentAge>

 <owl:Class rdf:ID="Employee"/>

 <owl:ObjectProperty rdf:ID="employmentAge">
    <rdfs:domain rdf:resource="#Employee"/>
    <rdfs:range rdf:resource="#EmploymentAge"/>
 </owl:ObjectProperty>
```

List 1 conceptualization of data value range constraint in OWL

4.2 Software Agent-Based Knowledge Representation Approach

The second approach is to utilize software agent technology. Software agent technology has been in extensive discussion for many years but it is perhaps recently that it has been attracting much attention of exploitation in the emergence of the Semantic Web. Basically software agents are components in an application that are characterized by among other things autonomy, pro-activity and an ability to communicate [15]. Autonomy means that agents can independently carry out complex and long term tasks. Pro-activity means that agents can take initiative to perform a given task without human intervention. Ability to communicate means agents can interact with other agents or other components to assist to achieve their goals.

In this paper we implement software agents using JADE (Java Agent Development framework), an agent-oriented middleware [16, 17]. The reason we use JADE is simply because it facilitates development of complete agent-based applications and it is written in well known object-oriented language, Java. More details of JADE can be found on its website (http://jade.tilab.com).

Basically in this paper, we utilize JADE agent technology to help define the knowledge of calculation. A JADE agent is identified under FIPA specifications [18] by an agent identifier. A task can be defined for an agent to carry out. Agent action defines the operations to be performed. Agent communication according to FIPA specifications [18] is the most fundamental feature of software agents. Format of messages is compliant with that defined by FIPA-ACL message structure.

For the calculation knowledge described in section 3.2, we can define it in the following formula.

```
Total Cost = Price * Quantity * (1 + GST Rate / 100)
```

Ontologies are typically specific to a given domain. For the above formula we specify to a product trading domain which would not be the same as in a payroll system. Thus product concept could have properties of name, barcode, etc. Agents then have some shared understanding with the product concept and its properties. There may be two products named the same. In order to unequivocally identify a product, it may be necessary to specify barcode.

According to the FIPA specifications [18], when agents communicate, product information representation is embedded inside ACL messages. Because JADE agents are Java-based, the information can be represented using objects.

In order to exploit agent and ontology technology to support and allow agents to discourse and reason about facts and knowledge related to a given domain, we specify the approach into 3 steps.

- Define concepts in an ontology. In the purchase example, it includes *Product* and *Purchase* concepts.
- Develop proper Java classes for the above two concepts in the ontology.
- Define the calculation formula by hard-coding it.

In order to illustrate defined concepts of *Product* and *Purchase* in an ontology, we model *Product* and *Purchase* knowledge representation shown in Figure 1. Figure 1 (A) shows *Product* concept and Figure 1 (B) shows *Purchase* concept. Ontology class *Product* has datatype properties of *name* and *barcode* both related to a string type. Ontology class *Purchase* has object properties of *item* related to the ontology class *Product*. The ontology class *Purchase* also has datatype properties of *price* related to a float type and *quantity* and *tax_rate* related to an integer type.

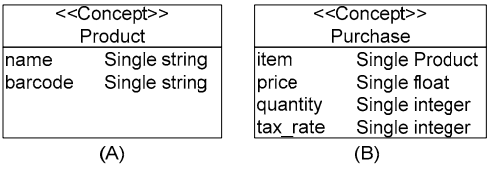


Figure 1 Product and Purchase concepts in ontology modelling

We reuse schema classes available in JADE *PredicateSchema*, *AgentActionSchema*, and *ConceptSchema* included in the *jade.content.schema* package to define the structure of each type of predicate, agent action, and concept respectively [17]. In the example, we can model the domain including one concept (Product), one predi-

cate (Purchase – to apply to a product) and one agent action (Calculate – to calculate total cost including tax).

Since the ontology is shared among agents, *TradeOntology* class is placed in an ad-hoc package, ontology. The ontology defined in Java is shown in List 2.

```
package TradingPackage;

import jade.content.onto.*;
import jade.content.schema.*;
import jade.util.leap.HashMap;
import jade.content.lang.Codec;
import jade.core.CaseInsensitiveString;

public class TradeOntology extends jade.content.onto.Ontology  {
  //NAME
  public static final String ONTOLOGY_NAME = "Trade";
  // The singleton instance of this ontology
  private static ReflectiveIntrospector introspect = new Reflec-
tiveIntrospector();
  private static Ontology theInstance = new TradeOntology();
  public static Ontology getInstance() {
     return theInstance;
  }

   // VOCABULARY
    public static final String PURCHASE_ITEM="Item";
    public static final String PURCHASE_QUANTITY="Quantity";
    public static final String PURCHASE_TAX_RATE="Tax_Rate";
    public static final String PURCHASE_PRICE="Price";
    public static final String PURCHASE="Purchase";
    public static final String CALCULATOR="Calculator";
    public static final String CALCULATE="Calculate";
    public static final String PRODUCT_NAME="Name";
    public static final String PRODUCT_BARCODE="Barcode";
    public static final String PRODUCT="Product";

  /*  Constructor   */
  private TradeOntology(){
    super(ONTOLOGY_NAME, BasicOntology.getInstance());
    try {

    // adding Concept(s)
    ConceptSchema productSchema = new ConceptSchema(PRODUCT);
    add(productSchema, TradingPackage.Product.class);

    // adding AgentAction(s)
    AgentActionSchema calculateSchema = new AgentAction-
Schema(CALCULATE);
    add(calculateSchema, TradingPackage.Calculate.class);

    // adding AID(s)
    ConceptSchema calculatorSchema = new ConceptSchema(CALCULATOR);
    add(calculatorSchema, TradingPackage.Calculator.class);

    // adding Predicate(s)
    PredicateSchema purchaseSchema = new PredicateSchema(PURCHASE);
    add(purchaseSchema, TradingPackage.Purchase.class);

    // adding properties
    productSchema.add(PRODUCT_BARCODE, (Term-
Schema)getSchema(BasicOntology.STRING), ObjectSchema. MANDATORY);
    productSchema.add(PRODUCT_NAME, (Term-
Schema)getSchema(BasicOntology.STRING), ObjectSchema.OPTIONAL);
    purchaseSchema.add(PURCHASE_PRICE, (Term-
Schema)getSchema(BasicOntology.FLOAT), ObjectSchema.MANDATORY);
```

```
      purchaseSchema.add(PURCHASE_TAX_RATE, (Term-
Schema)getSchema(BasicOntology.INTEGER), ObjectSchema.MANDATORY);
      purchaseSchema.add(PURCHASE_QUANTITY, (Term-
Schema)getSchema(BasicOntology.INTEGER), ObjectSchema.MANDATORY);
      purchaseSchema.add(PURCHASE_ITEM, productSchema, ObjectSchema.
MANDATORY);

     }catch (java.lang.Exception e) {e.printStackTrace();}
    }  }
```

List 2 Trade Ontology defined in Java

```
package TradingPackage;

import jade.content.*;
import jade.util.leap.*;
import jade.core.*;

public class Product
      implements Concept {

// Barcode
 private String barcode;
 public void setBarcode(String
value){
       this.barcode=value; }
```

```
public String getBarcode() {
    return this.barcode;
}

// Name
 private String name;
 public void setName(String value) {
      this.name=value;
 }
 public String getName() {
      return this.name;
 }
}
```

List 3 Product concept defined in Java

```
package TradingPackage;

import jade.content.*;
import jade.util.leap.*;
import jade.core.*;

  public class Purchase imple-
ments Predicate {

  // Price
   private float price;
   public void setPrice(float
value) {
     this.price=value;
   }
   public float getPrice() {
     return this.price;
   }

  // Tax_Rate
   private int tax_Rate;
   public void setTax_Rate(int
value) {
     this.tax_Rate=value;
    }
```

```
   public int getTax_Rate() {
     return this.tax_Rate;
   }

  // Quantity
   private int quantity;
   public void setQuantity(int
value) {
     this.quantity=value;
   }
   public int getQuantity() {
     return this.quantity;
   }

  // Item
   private Product item;
   public void setItem(Product
value) {
     this.item=value;
   }
   public Product getItem() {
     return this.item;
   }
}
```

List 4 Purchase concept defined in Java

The schemas for *product*, *purchase*, *calculate*, and *calculator* concepts are associated with *product.java*, *purchase.java*, *calculate.java*, and *calculator.java* classes respectively. Each property in a schema has a name and a type. For example, in the *product* schema, *barcode* has its type as string. Every product must have barcode as

declared as MANDATORY. Similarly, value for properties *item*, *price*, *quantity*, and *tax rate* cannot be null because when the purchase is made these values are mandatory. Validation is made by throwing an exception if the value of mandatory properties is null.

The *product* concept could be defined specifically to particular products e.g. books, CDs for more specific trading. Properties of the product concept i.e. *name* and *barcode* will be inherited to books and CDs. Book and CDs concepts can have their own specific properties e.g. the CDs concept might have *tracks* property and books might have *authors* property and so on.

Java classes, associated with the product concept and the purchase predicate in the example, are shown in List 3 and List 4 respectively.

Agent action associates with the agent identifier which is intended to perform action for this example to calculate total cost included tax. Calculation can be hard coded getting value from object of class purchase i.e. price, quality, and tax rate.

For example a product of $200 price, 2 quantity, and 10% tax rate would have expression as following:

```
((action (agent-identifier :name calculator) calculate (product :name "xxx" :barcode "01211") purchase (product :name "xxx" :barcode "01211") 360)
```

Alternatively, we can also specify in class purchase as the attribute of *TotalCost* shown as in List 5 below.

```
// Total Cost
   private float TotalCost;
   public float getTotalCost() {
     return this.price * this.quality * (1 + tax_Rate / 100);
}
```

List 5 The formula defined in Java

One advantage using software agent based approach, in comparison to the conceptualization approach, is that it has already been realized in software agent definition which does not require further implementation code.

5 Conclusion

In this paper we addressed the practical problems associated with knowledge representation in OWL. OWL specifications provide many mechanisms for defining restrictions and associations among *Classes* but not for *Properties*. We have presented two types of knowledge, which are common to various domains, but cannot be modeled directly using constructs specified in OWL. To tackle this knowledge presentation gap in OWL, we have proposed two alternative solutions to the problems. One is to conceptualize the knowledge such as the data value range constraints and the other is to use other existing technology such as software agents to encode and convey the

knowledge. As we gave calculation knowledge example defined in the formula for experimentation. With the knowledge defined in the ontology the software agents are able to use calculation knowledge to define a new knowledge (i.e. the total cost). Its prototype is still under development and need to extend in different fields. We do not intend to list all OWL modeling problems rather we aim to provide some useful hints to other likewise knowledge representation issues with OWL that have yet to be resolved.

References

1. R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," Data & Knowledge Engineering, vol. 25, pp. 161-197, 1998.
2. W. Borst, "Construction of Engineering Ontologies," University of Twente, Enschede, 1997.
3. T. R. Gruber, "A Translation Approach to Portable Ontology Specifications," Knowledge Acquisition, vol. 5, pp. 199-220, 1993.
4. V. Kashyap, "Design and creation of ontologies for environmental information retrieval," presented at the 12th Workshop on Knowledge Acquisition, Modeling and Management, Alberta, Canada, 1999.
5. L. Stojanovic, N. Stojanovic, and R. Volz, "Migrating data-intensive Web Sites into the Semantic Web," presented at the 17th ACM symposium on applied computing (SAC),, SAC, 2002.
6. R. Meersman, "Ontologies and Databases: More than a Fleeting Resemblance," presented at OES/SEO Workshop Rome, Rome, 2001.
7. I. Astrova, "Reverse engineering of relational database to ontologies," presented at First european Semantic Web symposium, ESWS, Heraklion, Crete, Greece, 2004.
8. M. LI, X.-Y. DU, and S. WANG, "Learning ontology from relational database," presented at the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, China, 2005.
9. S. Zhao and E. Chang, "Mediating Databases and the Semantic Web: A methodology for building domain ontologies from databases and existing ontologies," presented at SWWS'07- The 2007 International Conference on Semantic Web and Web Services, Las Vegas, Nevada, USA, 2007.
10. W3C, "Ontology Web Language," vol. 2006, M. K. Smith, C. Welty, and D. L. McGuinness, Eds.: WC3, 2004.
11. W3C, "OWL Web Ontology Language Use Cases and Requirements," vol. 2007, J. Heflin, Ed., 2004.
12. N. Shadbolt, W. Hall, and T. Berners-Lee, "The Semantic Web Revisited," IEEE Intelligent Systems, pp. 96-101, 2006.
13. D. Reynolds, C. Thompson, J. Mukerji, and D. Coleman, "An assessment of RDF/OWL modelling," Digital Media Systems Laboratory, HP Laboratories Bristol 28 Oct 2005.
14. W3C-RIF, "Rule Interchange Format Working Group," vol. 2007: W3C, 2007.
15. M. Wooldridge, Introduction to MultiAgent Systems. , 1st ed: John Wiley & Sons., 2002.
16. F. Bellifemine, "JADE Java Agent DEvelopment Framework," Telecom Italia Lab: Torino, Italy., 2001.
17. F. Bellifemine, G. Caire, and D. Greenwood, Developing Multi-Agent Systems with JADE: John Wiley & Sons Ltd., 2007.
18. F. Bellifemine, A. Poggi, and G. Rimassa., "JADE: a FIPA2000 compliant agent development environment," presented at The fifth International Conference on Autonomous Agents, Montreal, Quebec, Canada, 2001.

REPRESENTATION, REASONING AND SEARCH

Context Search Enhanced by Readability Index

Pavol Navrat, Tomas Taraba, Anna Bou Ezzeddine, and Daniela Chuda[1]

Abstract Context search is based on gathering information about user's sphere of interest before the search process. This information defines context and augments search query in subsequent phases of search to attain better search results. There are several basic methods for context enhanced searching. The main idea of them is to extract keywords of the found document and compare them with those from the context. The keyword recognition process is difficult to describe in a formally complete way. The context search based on it may, but also may not attain better search results. We propose a modification of the context search by broadening the scope of kinds of attributes, i.e. to consider also implicit attributes rather than only keywords (i.e., explicit ones). Our hypothesis is that it will enable the context search method to fetch more relevant results. This work analyzes the relation between readability index of a document and its content. Improvement idea is based on the kind of knowledge which is difficult to express by keywords, e.g., the fact that user is looking for fairy tales rather than science articles.

1 Introduction

A person formulating search query on the web is doing this with some context in mind. Let us call him or her the interested person (IP), since he or she is interested in some specific information at one moment.

In many cases IP works with other documents, files or web pages. Information gathered from these documents can be used to find out the IP's actual scope of interest. This information can be stored in some specific structure and then used to receive better search results.

Many ways to augment search query by context can be explored, but the common base of them seems to be using document keywords. In several related

[1] Slovak University of Technology, 842 16 Bratislava, Slovak Republic email: navrat@fiit.stuba.sk, tomastaraba@wms.sk, ezzeddine@fiit.stuba.sk, chuda@fiit.stuba.sk

Please use the following format when citing this chapter:

Navrat, P., Taraba, T., Ezzeddine, A.B. and Chuda, D., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 373382.

works [1, 2] keywords are used to rate the relation between search result and context using ontology of keywords [3], or simply by using keywords in context vector [1]. Some experiment with semantic query expansion [4]. Keywords from context can be inserted into search query string and sent to standard search engine. Another approach is to submit the original query first, and then reorder the resulting documents according to the rate how they fit the context. There can be made various combinations of query and keywords from context, send them to multiple search engines and then aggregate several sets of results.

Each kind of method may have advantages and disadvantages, but in some sense they are similar. We propose, however, to broaden the concept of context. We modified the keyword-based approach hypothesizing that considering also other attributes rather than only keywords can result in better search precision [5]. We made series of experiments with on-line database and verified that it tends to attain better search results when using the multi-attribute context.

While the experiments have shown that our idea works quite fine with an on-line database where many attributes are present [6], we were not sure how this can be used in the web. There are not so many attributes identifiable in the web like in some on-line database. For example, in the online database we tracked the author of each document in context. We assumed that the name of an author is an important information. It can be quite safely assumed that one author writes on a small set of themes, and one theme is written about by a not so big set of authors. But consider a web page. It is hard to define the exact process to extract the name of the author of each webpage; in most cases we can tell that extraction is impossible.

The rest of the paper is structured as follows. In Section 2 - Motivation we outlined the motivation why we have some concerns about classic context search based on keywords. We formulated a sample problem and described it. The proposed approach to solve the problem, which is based on our improvement idea is described in Section 3 - The Proposed Modification. In section 4 we formulated a hypothesis and attempted to verify it by series of experiments. The results and their consequences have inspired us to proceed in the research, concentrating mainly on readability index values. Their possible interpretation is described. Having gathered sufficient research results, we were able to draw a conclusion and we suggest some future work in section 5 - Conclusion.

2 Motivation

As a motivating example for our work, let us consider this problem: Suppose IP looking for fairytales on the web. IP read two documents. The first one was Little Red Riding Hood, and the second one The Wandering Egg.

There are some concerns about the relevance of the classical context search. Ensuring relevance in this case means to get results of documents containing fairy

tales for children under 5 years instead of getting results of Cookery-book for hunters, which can contain more keywords “egg, roe, hunter, food” than a common fairy tale.

Troubles With Keywords. When one compares two different fairy tales, it can be quite hard to find at least a few common keywords. For example, let us consider Little Red Riding Hood in the context of having keywords: little, grandmother, wolf, wood, door, hunter, etc. In our context search with the query Cinderella, the approach is to prefer search results containing fairy tale with keywords: prince, girl, time, sisters, shoe, three nuts, etc. rather than documents about Cinderella band containing keywords: band, tour, metal, rock, album, etc.

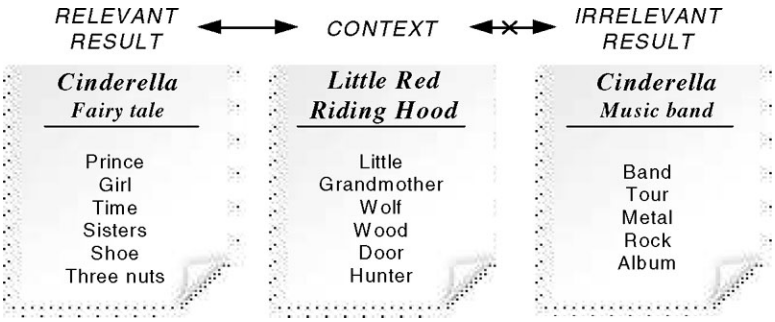


Fig. 1 Comparison of relevant and irrelevant result keywords

As one can see in Fig. 1, there is no relation between the two fairy tales with regards to keywords, because each one tells a different story. Keywords do not tell us that Cinderella story is relevant to Little Red Riding Hood as both are fairy tales and Cinderella band presentation isn’t relevant because it is not a fairy tale.

Improvement Idea. Going out from previous work described in introduction, the improvement idea is to use some combination of explicit and implicit attributes to rank the search results. We introduce implicit attributes which can help us to guess whether the found document is fairy tale, romantic novel, presentation of the band, advertisement, scholarly article, etc.

If we were in an on-line database, and we had the name of the author, we could use it. Since only a relatively small number of authors write fairy tales, the author attribute can be effective in search. But on the web, there is usually no such explicit attribute available. Unfortunately, we cannot guarantee that every document has the name of the author included in it somewhere. Besides that, there is also not an explicit attribute in every text that would inform us how the text has been written.

Flesch Readability Index. The improvement idea is based on a formula defined by Flesch [7]. Let us consider user’s age and level of ability to read and comprehend the text. There is a method to categorize readability of a given text by

the Flesch readability index (FRI). This method is used for estimating the reading comprehension level necessary to understand a written document. For a given document, the Flesch readability index is an integer (0-100) indicating how difficult the document is to understand, with lower numbers indicating greater difficulty.

$$FRI = 206.835 - (1.015 x \frac{number of\ words}{number of\ sentences}) - (84.6 x \frac{number of\ syllables}{number of\ words}) \quad (1)$$

Flesch categorized readability indexes into 7 educational levels and describe Flesch Reading easy scale. In 1948, Flesch published [7] the results of his study of the editorial content of several magazines and he found that about 45% of the population can read The Saturday Evening Post, nearly 50% of the population can read McCall's, Ladies Home Journal, and Woman's Home Companion, slightly over 50% can read American Magazine and 80% of the population can read Modern Screen, Photoplay, and three confession magazines. For example comics have readability index 95, New York Times 39, Auto Insurance 10.

What Improvement Do We Suggest*?* Let us consider that IP has never read a document classified as more difficult than a document comprehensible by a high school student. Is there a reason to return him also documents understandable solely by a law school graduate? In our example, why to return e.g. academic analytical studies on the Cinderella fairy tale, or why to return lyrics of the Cinderella music band songs, or why to return any other documents for which the readability index indicates that they are not fairy tales?

3 The Proposed Modification

To develop our improvement idea, it should be incorporated in some existing context search method. The improvement means essentially adding more attributes into context and considering more attributes in the process of result selection. A suitable way is to change the ranking function of the Rank-Biasing method [1]. In general, this method uses context to change the score of every search result and then sorts the results by the new score. Finally, the new set of results will contain all the results from original set (as if the context was not used) but the results are sorted in a new order with more relevant results on the top. The relevancy of result is determined by how much the result fits the context – how many keywords from the context are contained in the search result.

Our modification proposes to change the rank function. We use a context of keywords and a context of readability index values. The relevancy of a result depends on how much the result fits the keywords in the context, but also on how much it fits the readability indexes in the context.

Context Acquisition. The context is acquired while the user is browsing [2]. For example, we can acquire it by tracking every click (i.e, loading of a document

specified by URL) which the IP does and store the information from the tracked documents in the context. The context is represented by a vector, which contains two kinds of dimensions: dimensions of keywords and dimensions of readabilities. A dimension is represented by a vector of attribute values (attributes being either of keyword or readability index kind). Each value has a score, which determines how many times the keyword occurred in all documents of context, or how many documents had the given readability index:

$$C = \begin{Bmatrix} D_1 \\ D_2 \\ \dots \\ D_N \end{Bmatrix}, \qquad D_i = \{(v_1 \to s_1),(v_2 \to s_2),\dots,(v_j \to s_j),\dots,(v_M \to s_M)\}, \tag{2}$$

where C is context vector, D_i is a dimension of the context vector, v_j is a value of an attribute represented by the given dimension, s_j is score of value v_j.

Search Process. The search process is initiated by a search query sent by IP. First, the search query is sent to a standard search engine to receive a set of results. Next, the ranking score is modified for each result from the result set. The ranking score given by the standard search engine is a number indicating how much the result fits the query. We modify it to indicate how much the document fits not only the query, but also the context. Having modified the score of each result, the result set will be sorted.

Ranking Function. We propose in our modification to change the ranking function. While the original ranking function calculates the rank score only using context of keywords, the modified function calculates it by considering also the context of readabilities. There are several possibilities how to combine the two rankings. In [3], an additive formula is used, but we found it more useful to use a multiplicative one. The final score is calculated by the formula $R = R' * R_K * R_R$. In this formula, R' is the original score assigned by the standard search engine, R_K is the rank factor calculated when ranking by keywords and R_R is rank factor calculated when ranking by readability index. For every rank factor we require it to have values from $\langle 1,2 \rangle$ interval, so every value has to be mapped into this interval.

Ranking by Attribute of Keywords. The rank factor for an attribute of keywords says how much the keywords in document fit the keywords in context. In detail we have two sets: set of document keywords and set of context keywords. We require the rank factor to have value of 2 if every context keyword is found in the given document and have value of 1 if no context keyword is found in the given document.

$$R_K = 1 + \frac{sum(score\_of\_every\_context\_keyword\_found\_in\_document)}{sum(score\_of\_every\_context\_keyword)} \tag{3}$$

Ranking by Readability Index Attribute. A readability index attribute has a numeric value. In the process, the relevant document is the document, which has the readability index value within the specific interval. This interval is determined by readability index values in context. For example - if the readability index vector in the context is: {70, 72, 74, 76}, then the interval would be <70; 76>.

The rank factor for the readability index attribute has the value of 2 when the readability index of given document is in the centre of the interval and value of 1 when the value is outside far from the borders of the interval.

4 Experimental Testing

We performed series of experiments to test the improved method. We wanted to see how ranking by readability influences the precision of the search. We devised three cases and performed simulations in these three areas of user's interest:

Fairy Tales. Relevant results are pages containing text of a searched fairy tale. Among irrelevant results, we also marked pages containing information on movies, bookstore catalogues, and other.

Population Diseases. We searched for documents describing a given population disease. As relevant we marked pages containing statistical studies related to the given population disease, scholarly articles, articles on research in this area, popular articles explaining the terms related to given disease. As irrelevant we marked pages presenting, propagating, or selling medicaments, presentations of health organizations, centers, and founds.

Predators. As relevant we marked pages containing some information about the given predator, group of predators or presentations of Zoo's. As irrelevant we marked pages using the name of predator in meanings other than that of an animal (e.g., a car, computer, etc.), presentation of companies using the name of the predator as the brand, pages of conservation organizations.

For each area we collected hundreds of results by series of queries and marked them as relevant or irrelevant. Then we added each relevant document into context in successive steps simulating user's clicks on relevant documents. After each addition we recalculated score of each result, sorted a result set and calculated the precision of the search for every configuration of context vector.

Results. As we tested the method in three different areas, we have three different experimental results. We tracked how the precision has changed considering the change of context size.

In all the figures, the KW curve represents the precision of method using only keywords in ranking, the curve marked as RI represents precision of method ranking only by readability index attribute. The curve marked as KWRI represents the combination of both kinds of attributes aggregated together. SS represents the precision of standard search engine. Precision of the standard search engine does not change when the context size grows.

In the first case – searching fairy tales – there is a very significant improvement. The modified method produces more precise results than the original method. As we can see in Fig. 2, major improvement is caused by using the readability index attribute in ranking.

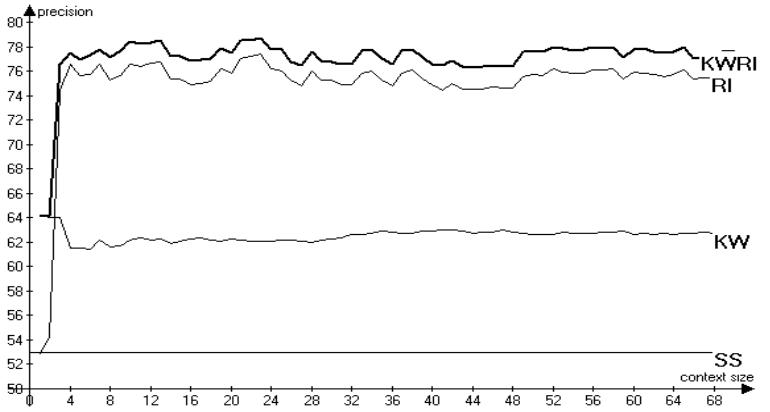


Fig. 2 Precision in the case of fairy tales

In the second case – population diseases – the modified method produces more precise results, but it does not yield a significant improvement (Fig.3). The average difference in precision between the original and modified method is around 3 - 4%.

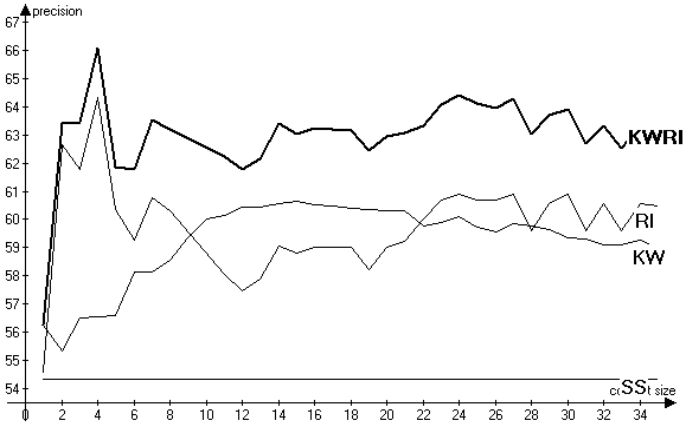


Fig. 3 Precision in the case of diseases

In the third case – predators in the nature – the modified method produces more precise results most of the time, but sometimes the precision could be worse than that of the original method (Fig. 4). It was caused by adding documents very different in readability index attribute into context at context size of 30. On the other side, method using only keywords in rank function keeps on the same precision during the context growth. It may mean that keywords are nearly the same and adding each new document into context does not strongly influence the vector of keywords. In combination of both methods, the precision is sometimes better than the precision of the original method (KW), but sometimes the search produced worse results.

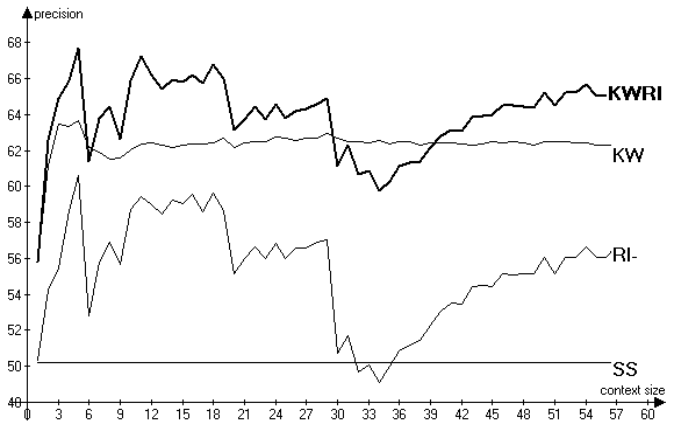


Fig. 4 Precision in the case of predators

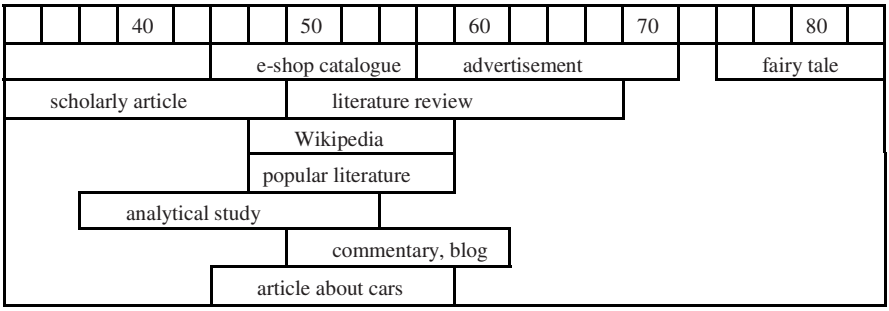
In all three figures we can see the cold-start problem. The RI method does not yield good results when the size of the context is small. While the KW method works fine with 1 or more documents in the context, the RI method requires having a minimum of 2 documents in the context. This is caused by readability index value interpolation with requirement minimum of 2 values to find the interval border values.

Additional Research: What Was Wrong? As we can see in the graph representing experiment results, the method works fine for searching fairy tales, but does not work so well for the other areas. We tried to find out where the problem is and why it occurs.

After a small analysis of acquired pages, we found out the problem. We manually classified the type of the content of every ranked page provided as search result. We classified 1,000 pages into 10 groups of content type and then statistically determined the interval of readability index values for each group. Results of this experiment are represented in Table 1.

The conclusion is - Fairy tales are very special documents. Their readability index is very high due to the simplicity of the text. In Table 1 we can see that fairy tales are sufficiently well separated from other documents. The case with the popular literature on predators is not similar, which is overlapped by several other different genres (articles about cars, advertisements, e-shop catalogue, blogs). We conjecture this is the fundamental reason why fairy tales are easily identifiable by the readability index but e.g. popular articles on predators are not.

Table 1 Overlapping of the readability index values for different types of text



A popular article on a predator is too common in comparison to a fairy tale, or a scholarly article, which are quite specific kinds of texts. In the table above we can see that very common and ordinary documents have readability index values around the value of 50. Experiments show the average value of the readability index is 48. Based on this research we assume that major “common” documents have readability index around 50. Other documents with much higher or much lower values are more “specific”.

5 Conclusion

This work is focused on investigation in the relation between readability index and the character (related also to elements of style, genre) of the web page. We tried to find out how the readability index can be used to attain better search results in context search.

What Did We Find Out? In general, our improvement idea works fine, but the degree of improvement depends on the type or character of documents. Readability index is related to the character of the text, but generally it may not be sufficiently restrictive to allow the desired identification of the sphere of interest of the query in all cases. Relying on the readability index in the context search was not effective when the user is interested in very common things. It is particularly effective when the IP is interested in not so common types of texts, e.g. fairy tales

or scholarly articles. In those cases we can quite safely tell that documents have similar content and the user has some specialized sphere of interest.

Future Research Work. There may be other ways to overcome the problem of ordinary texts. When dealing with our motivating problem, there is no need to consider the interval in which the readability index belongs. A better indication may be the distance of readability index of current document from the "general centre". In other view, in the context the readability index measures how different the documents of the context are from the „ordinary ones". As it can be seen from the table of readability index of different types of web pages, ordinary documents are pages with the readability index around the value of 50. The more specific the document is, the bigger is the distance between its readability index and the value of 50.

ACKNOWLEDGMENTS

This work was partially supported by the Slovak State Programme of Research and Development "Establishing of Information Society" under the contract No. 1025/04 and the Scientific Grant Agency of Republic of Slovakia, grant No. VG1/3102/06.

References

1. Kraft R., Chang C.C., Maghoul F., Kumar R.: Searching with Context. In Proceedings of the 15th International Conference on World Wide Web WWW '06. Edinburgh, pp. 477- 486, ACM Press, (2006).
2. Bharat K.: SearchPad: Explicit Capture of Search Context to Support Web Search. In Proceedings of the 9th International World Wide Web Conference, pp. 493-501, Elsevier, Amsterdam (2000).
3. Challam V., Gauch S., Chandramouli A.: Contextual Search Using Ontology-Based User Profiles. In Proceedings of the 8th Large-Scale Semantic Access to Content Conference (RIAO'2007), Pittsburgh (2007).
4. Malecka J., Rozinajova V.: An Approach to Semantic Query Expansion. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. Research Project Workshop Proceedings, pp. 148-153, STU, Bratislava (2006).
5. Clarke S., Willett P.: Estimating the Recall Performance of Search Engines. ASLIB Proceedings, 49 (7), pp. 184-189 (1997).
6. Navrat P., Taraba T.: Context Search. In: Y. Li, V.V. Raghavan, A. Broder, H. Ho: 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (Workshops), Silicon Valley, USA, pp. 99-102, IEEE Computer Society, (2007).
7. Flesch R.: A New Readability Yardstick. Journal of Applied Psychology, Vol. 32, pp. 221-233, (1948).

Towards an Enhanced Vector Model to Encode Textual Relations: Experiments Retrieving Information

Maya Carrillo[1] and A. López-López[2]

Abstract. The constant growth of digital information, facilitated by storage technologies, imposes new challenges for information processing tasks, and maintains the need of effective search mechanisms, oriented towards improving in precision but simultaneously capable of producing useful information in a short time. Hence, this paper presents a document representation to encode textual relations. This representation does not consider each term as one entry in a vector but rather as a pattern, i.e. a set of contiguous entries. To deal with variations inherent in natural language, we plan to express textual relations (such as noun phrases, named entities, subject-verb, verb-object, adjective-noun, and adverb-verb) as composed patterns. An operator is applied to form bindings between terms encoding relations as new "terms", thereby providing additional descriptive elements for indexing a document collection. The results of our first experiments, using the document representation to conduct information retrieval and incorporating two-word noun phrases, showed that the representation is feasible, retrieves, and improves the ranking of relevant documents, and consequently the values of mean average precision.

1 Introduction

The increment of information in digital form over the last decade imposes new challenges for information processing tasks, such as: topic detection and tracking, clustering, information retrieval, question answering, or classification. The success

[1] Maya Carrillo

Instituto Nacional de Astrofísica Óptica y Electrónica, Facultad de Ciencias de la Computación, BUAP, email: cmaya@inaoep.mx

[2] A. López-López

Instituto Nacional de Astrofísica Óptica y Electrónica, Luis Enrique Erro #1 Santa María Tonantzintla, 72840 Puebla, México, email: allopez@inaoep.mx

Please use the following format when citing this chapter:

Carrillo, M. and López-López, A., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 383392.

of these tasks depends on how well the language can be modeled and expressed in the computer. In practice, deep language understanding has remained elusive, while the "bag of words" model continues to prevail in information processing tasks. In particular, the classic information retrieval (IR) techniques rest on the assumption that if a document and a query have a word in common, then the document is about the query. If the number of words in common increases, the relation is stronger. Under this assumption, the IR problem is reduced to determine to what extent the bag of keywords in the user's query matches those representing the documents. This approach is widely used since it quickly generates acceptable results. However, it does not consider linguistic phenomena such as: morphological variation, which originates words with different number, gender, and tense; lexical variation, where different words have the same meaning; syntactical variation, where word order changes meaning; and semantic variation, where a single word has multiple meanings. The language is more than simply a collection of words. Rather, it is used to refer to entities, concepts and relations that are expressed in grammatical forms. For example, with word order; v*enetian blind* does not mean the same as *blind venetian*. Moreover, words are combined in phrases and larger structures which remain joined by relations such as: structural dependencies, coreferences, semantic roles, speech dependency, intentions, and others. Based on the previous considerations, it has been conjectured that a more suitable text representation would have to include groups of words like phrases or expressions that denote meaningful entities, concepts, or relations within the search domain. Some phrase extraction methods use syntactical analysis and try to capture semantic uniformities from the superficial structure, approaching content to some degree. Syntactical phrases seem to be reasonable content indicators, since they allow identifying change in the word order. However, this syntactical analysis is far from a real semantic analysis. Researchers working in the area have used techniques of natural language processing (NLP) to do IR, supposing that a better understanding of the request and document information is the key to improve the retrieval effectiveness.

In this paper, we propose an enhanced vector document representation that considers a document to be the sum of its term-vectors. It uses circular convolution operator to encode relations between terms. The document representation has been used to define an information retrieval model and the experiments carried out have showed that the model is capable of retrieving documents, which are relevant to a user. The precision level is equivalent to that obtained with the classical vector model, but the enhanced model has the potential to allow the encoding of noun phrases, and hence other relations, to improve precision.

The remainder of this paper is organized as follows: Section 2 provides a brief description of related work, particularly on information retrieval. Sections 3 and 4 describe our proposed representation and retrieval model. Conclusions and future work are summarized in Section 5.

2 Related Work

Defining new models and focusing IR from different perspective extend the knowledge within the area. In the following paragraphs, previous works that emphasize the interest to establish new information retrieval models are described.

There are several previous works, suggesting the use of more than mere simple terms to index and retrieve documents. For instance, Lewis & Sparck Jones [5] suggest that appropriate strategies for document retrieval could be extended to allow well-motivated compound terms and similar descriptive units. They established that there are two main challenges for NLP technologies in IR: first, in making these technologies operate efficiently and effectively on the necessary scale, and second, in conducting the evaluation tests that are essential to discover whether the approach works.

Evans & Zhai [2] present an approach to index noun phrases for IR. They describe a hybrid method to extract meaningful (continuous or discontinuous) sub compounds from complex noun phrases. Their results improve both recall and precision.

Mitra *et al* [6] present a study that compares the usefulness of phrase recognition by using linguistic and statistical methods. They conclude that phrases are useful at lower ranks of precision when connection between documents and relevance is minimal, as long as a good ranking scheme is defined.

Regarding the recent proposals of new retrieval models, Shi *et al* [8] propose the Gravitation-Based Model (GBM), a model of IR inspired by the Newton's theory of Gravitation. In this model, a term is defined as a physical object composed of particles with a specific form (sphere or ideal cylinder) that has three attributes; type, mass, and diameter. Two particles of the same type are mutually attracted. A document and a query are modeled as a list of terms. Their total mass is calculated as the sum of the masses of all its constituent terms. The relevance of a document given a query is calculated as the attraction force between the objects corresponding to them.

Gonçalves *et al* [3] present a model that enhances traditional vector space model, establishing co-occurrence relations between named entities. They identify these named entities and determine the strength of co-occurrence relations among them, based on the distance that separates the entities and on the co-occurrence relation frequency. Given a document D where entities e_1, e_3, e_4, e_5 appear, if by the corpus analysis, it is known that e_1 has a strong co-occurrence relation with entity e_2, then when forming the vector D, e_2 is added. The cosine between the expanded vector of each document and the vector of a term-based query is used to rank documents. The method is evaluated using F measure to compare it against four standard statistical methods in IR: mutual information, Phi-squared, Vechtomova Mutual Information, and Z score. In all cases, the results obtained with the extended model are improved. The experiments were done using the CISI collection.

Becker & Kuropa [1] present a Topic-based Vector Space Model (TVSM), to compare documents regarding their content. They consider a *d* dimensional positive vector space *R*, where each dimension represents an orthogonal topic with respect to the others (e.g. *literature*, *computation*). A term vector (*software*, *program*) related to a topic points to the same direction as that topic (*computation*). A document is the sum of its term vectors multiplied by the frequency of each term in the document. The similarity between two documents is calculated as the scale product of document vectors. Finally, the authors do not report experiments concentrated on defining the theoretical model.

In addition to the continuous work that is being made in the area looking for new information retrieval models, it is important to mention some examples that show how textual relations have improved different performance levels in the systems. Thus, Vilares *et al* [9] have researched on retrieving information applying NLP techniques. The authors use tagged words to construct noun phrase trees, and their syntactic and morphologic variations. The constructed trees are embedded to obtain a syntactic pattern with all the binary dependencies (name-modifier, subject-verb and verb-complement) possible. This pattern is translated into a regular expression that preserves the binary dependencies, and allows extraction of multiword terms to index documents. The authors worked on the CLEF 2001/02. In the first set of experiments, both simple and complex terms are combined and indexed. A second set of experiments was done using syntactic information extracted from the documents, but not from the queries. The query is submitted to the system where the most informative dependencies of the top documents are selected to expand the query. Their results show improvement, which allow observing that the improvement even remains using only noun phrases, although to a lesser degree.

3 Representation and Similarity Assessment

Considerations done in sections 1 and 2 have led to our research question: What would be the impact on information processing tasks, if we consider relations among terms, associating them and using these associations as units to assess the similarity between documents? Working particularly on IR, the related work indicates that the success of applying NLP techniques has not been definitive. Our hypothesis is that the selected representation has influenced the success. Therefore, a vector representation with the potential to handle relations between terms is illustrated. This representation is inspired from previous efforts in cognitive science to explain how our brain processes analogies [7].

The traditional vector representation associates a single vector entry to each term, whose value is further made depending on its frequency. Our proposal represents a term by more than one vector entry, i.e. a short pattern formed by five binary contiguous digits and their corresponding positions in the whole vector. To

illustrate this concept, let's suppose a ten dimensional space and two entry patterns, if t_2 is a term whose pattern is defined as $[v_2,v_3]$ where subscripts indicate positions in the whole vector, i.e. the vector $\vec{t}_2$ representing only such term is: $\vec{t}_2 = [0,0,v_2,v_3,0,0,0,0,0,0]$. Since our proposal aims to express relations between terms, representing each term as a pattern inside a vector, allows encoding each intended relation according to the terms involved.

A document is represented adding the corresponding term vectors to form the document vector. Thus vector addition is used to represent documents and queries as a set of features. If D is a document whose terms are $t_1, t_2, \ldots t_n$, then its representation is: $\vec{D} = \langle \vec{t}_1 + \vec{t}_2 + \ldots + \vec{t}_n \rangle$ where the arrow on the literals, indicate that they represent vectors. After adding the terms, the document vector is normalized, denoted by $\langle ... \rangle$, and each term can be weighted according to its importance within the document using a weighting scheme such as tf.idf. Continuing with the example above if document D has terms t_1, t_2, t_3, whose vectors are: $[v_0,v_1,0,0,0,0,0,0,0,0]$, $[0,0,v_2,v_3,0,0,0,0,0,0]$, $[0,0,0,0,v_4,v_5,0,0,0,0]$ respectively, the vector representing D, without normalization is: $\vec{D} = [v_0,v_1,v_2,v_3,v_4,v_5,0,0,0,0]$.

However, vector addition is not enough to encode structure since it simply places together the features, whereas encoding structure requires a way to bind particular features together. For this purpose, we are using circular convolution as a binding operator to encode associations among term-vectors (i.e. structure).
Circular convolution maps two real-valued n-dimensional vectors into one. If x and y are n-dimensional vectors (subscripted 0 to $n-1$), then the elements of $z = x \circledast y$ are:

$$z_i = \sum_{k=0}^{n-1} x_k y_{i-k}$$

where subscripts are taken modulo-n and $\circledast$ denotes circular convolution. This binding operator keeps the same size of the vectors, can be decoded easily, preserves structural similarity, and is suitable for recursive application [8]. In addition to term-patterns, we have special patterns to identify the kind of relation and the "role" of the term involved (e.g. noun phrase right, noun phrase left, subject, verb, object, adjective, and adverb). These special patterns together with the term-patterns placed in their appropriate positions in order to build the corresponding vectors (term-vectors) are used to encode textual relations using the circular convolution operator. Given a relation $R(r_1, r_2)$ between terms r_1 y r_2, assuming they play a different role (i.e. the relation is non symmetric), to encode the relation two special patterns are needed: *left, right*. Then, the relation vector is:

$$\vec{R} = (\vec{left} \circledast \vec{r}_1 + \vec{right} \circledast \vec{r}_2)$$

where *left* (noun phrase left) and *right* (noun phrase right) allow us to distinguish between noun phrases like *venetian blind* and *blind venetian*.

Given a document D, with terms $t_1, t_2, \ldots, t_{x1}, t_{y1}, \ldots, t_{x2}, t_{y2}, \ldots, t_{xn}, t_{yn}, \ldots, t_n$, and relations R_1, R_2 among terms $t_{x1}, t_{y1}; t_{x2}, t_{y2}$, respectively its vector will be built as:

$$D = \langle \vec{t}_1 + \vec{t}_2 + \ldots + \vec{t}_n + (\vec{left} \circledast \vec{t}_{x1} + \vec{right} \circledast \vec{t}_{y1}) + (\vec{left} \circledast \vec{t}_{x2} + \vec{right} \circledast \vec{t}_{y2}) \rangle$$

Following the example and considering *D, t_1 ,t_2, t_3* as defined above, a relation between t_2 and t_3, and the special vectors $\vec{left} = [0,0,0,0,0,0,s_6,s_7,0,0]$ and $\vec{right} = [0,0,0,0,0,0,0,0,s_8,s_9]$, the circular convolution with $k = 0,..,9$ allows to combine the vectors defined to represent *D* having a relation between t_2 and t_3 as: $\vec{D} = \langle \vec{t}_1 + \vec{t}_2 + \vec{t}_3 + (\vec{left} \otimes \vec{t}_2 + \vec{right} \otimes \vec{t}_3) \rangle$. So, doing the operations:

$\vec{left} \otimes \vec{t}_2 = [s_7 v_3, 0,0,0,0,0,0,0,s_6 v_2, s_6 v_3 + s_7 v_2]$

$\vec{right} \otimes \vec{t}_3 = [0,0,s_8 v_4, s_8 v_5 + s_9 v_4, s_9 v_5, 0,0,0,0,0]$

$(\vec{left} \otimes \vec{t}_2) + (\vec{right} \otimes \vec{t}_3) = [s_7 v_3, 0, s_8 v_4, s_8 v_5 + s_9 v_4, s_9 v_5, 0, 0, 0, s_6 v_2, s_6 v_3 + s_7 v_2]$

$\vec{t}_1 + \vec{t}_2 + \vec{t}_3 = [v_0, v_1, v_2, v_3, v_4, v_5, 0,0,0,0]$

Thus, the vector of D without normalizing is as follows:

$\vec{D} = [v_0 + s_7 v_3,\ v_1,\ v_2 + s_8 v_4,\ v_3 + s_8 v_5 + s_9 v_4,\ v_4 + s_9 v_5,\ v_5, 0, 0,\ s_6 v_2,\ s_6 v_3 + s_7 v_2]$

In this way, if a document has the noun phrase *venetian blind,* its vector will include: $[... + (\vec{left} \otimes \vec{venetian} + \vec{right} \otimes \vec{blind}) + ...]$, a document with the noun phrase *blind venetian* will have $[... + (\vec{left} \otimes \vec{blind} + \vec{right} \otimes \vec{venetian}) + ...]$.

The dimension of the model vectors is relative to the number of vocabulary terms as the traditional vector model but increased by a constant factor (i.e. five). Term-vectors and relations-vectors are dynamically built when needed and only their patterns are stored.

A query has a similar representation. Our assumption is that the documents with these composed "terms" can be evaluated and ranked higher than those with only single terms.

Finally, to compute the similarity between queries and documents, we use dot-product. When the document vectors have relations encoded, the similarity is calculated as:

$$Sim = \left\langle \vec{d} + \delta * \sum_{j=1}^{m} f_j w_j \right\rangle \cdot \left\langle \vec{q} + \delta * \sum_{i=1}^{n} f_i w_i \right\rangle \quad (1)$$

Thus, the similarity is calculated as the dot product between two normalized vectors, built as the addition of a single term vector (i.e. $\vec{d}, \vec{q}$) and a relation vector multiplied by a factor (δ) less than one to ameliorate the impact of the coded relations.

4 Experiments

The proposed representation was applied to three traditional collections; CISI, CACM, and NPL, where CISI contains 1460 documents and 112 queries, CACM has 3204 documents and 64 queries, and NPL 11,429 documents and 93 queries. We selected these collections because they are well-known and relatively small to initially test our representation. The first produced a vocabulary of 5570 terms,

CACM had 5073 terms, and the third generated 7754 terms (after removing stop words and doing stemming).

The classical vector model was used as a baseline and implemented using tf.idf weighting. Cosine measure was used to assess similarity in the classical vector model. On the other hand, for the enhanced model, were defined as many term-patterns as vocabulary terms for each collection. In addition, documents and queries were represented using the vocabulary term-vectors combined by vector addition. Dot product was used as a similarity measure between documents and queries. The tf.idf weighting scheme was also used for our model. Our first experiment was aimed to test the feasibility of the representation, performing only term retrieval.

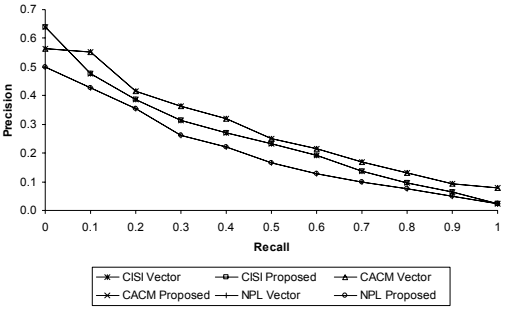


Fig. 1 Retrieval effectiveness on CISI, CACM and NPL using terms.

The customary recall-precision charts comparing our enhanced vector model against classical vector model are depicted in Figure 1. Precision was calculated at standard recall values averaged for the number of queries. The retrieval effectiveness in the classic vector model is equivalent to that obtained from the enhanced model for all collections: CISI, CACM and NPL, which is the reason why the curves are overlapped. These results show the feasibility of the representation and serve as a baseline for our further experiments. Our second experiment took into account a first relation between terms, in particular two-term noun phrases. We extracted noun phrases identified after parsing the documents and queries with Link Grammar [4], and selecting only noun phrases consisting of two contiguous words. After processing CISI collection, 8940 noun phrases were obtained, 9373 noun phrases for CACM and 18643 for NPL. The noun phrase vectors for each collection were calculated with the circular convolution operator applied to vectors involved. Since we used stemming to extract the vocabulary, we also kept the stems for the noun phrases. The same noun phrases were also added to the classical vector model as new terms. The tf.idf weighting scheme was used for both models.

The similarity between queries and documents that contain noun phrases for the enhanced model was calculated using (1).

Table 1. Recall-precision for queries with noun phrases.

Collection	Recall	Precision		% of Change
		Vector model/phrases	Enhanced model/phrases	
CISI (76 queries)	0	0.5871	0.6423	9.40
	0.1	0.4787	0.4797	0.21
	0.2	0.3849	0.3909	1.56
	0.3	0.3077	0.3151	2.40
	0.4	0.2636	0.2698	2.35
	0.5	0.2271	0.2344	3.21
	0.6	0.181	0.1912	5.64
	0.7	0.1319	0.1375	4.25
	0.8	0.0961	0.0973	1.25
	0.9	0.063	0.0641	1.75
	1	0.0242	0.0246	1.65
	Average	0.2496	0.2588	3.06
CACM (51 queries)	0	0.6099	0.5842	-4.21
	0.1	0.5580	0.5723	2.56
	0.2	0.4456	0.4292	-3.68
	0.3	0.3828	0.3709	-3.11
	0.4	0.3160	0.3162	0.06
	0.5	0.2422	0.2505	3.43
	0.6	0.2159	0.2159	0.00
	0.7	0.1709	0.1693	-0.94
	0.8	0.1340	0.1310	-2.24
	0.9	0.0942	0.0932	-1.06
	1.0	0.0801	0.0798	-0.37
	Average	0.2954	0.2920	-0.87
NPL (92 queries)	0	0.4430	0.5137	15.96
	0.1	0.3851	0.4421	14.80
	0.2	0.3044	0.3519	15.60
	0.3	0.2397	0.2590	8.05
	0.4	0.2060	0.2200	6.80
	0.5	0.1599	0.1665	4.13
	0.6	0.1301	0.1283	-1.38
	0.7	0.1038	0.0998	-3.85
	0.8	0.0782	0.0753	-3.71
	0.9	0.0501	0.0485	-3.19
	1.0	0.0239	0.0242	1.26
	Average	0.1931	0.2118	4.95

We summarize the recall-precision results for 76 queries (those actually having relevant documents) of CISI in Table 1. The precision improved in all standard re-

call levels, taking up to a 9.4% improvement and a 3.06% on average. Table 1 also presents the results for 51 queries of CACM where only three recall points are favorable for the enhanced model, even though the average difference is quite small at -0.87%. The same table shows the outcomes for 92 queries of NPL, where at seven recall points the data is favorable to our model, and only four are worse than those obtained with the classical vector model. The highest precision is reached in the first recall point having 15.96% of improvement. The average improvement for this collection was 4.95%.

We also used the mean average precision (MAP) and normalized precision (NPREC) metrics to compare the results. The MAP for 76 CISI queries was 0.2518 for classical vector model with noun phrases, and 0.2568 for our model, having an improvement of 3.42%. The MAP for CACM was 0.3155 for vector model and 0.3144 for our model, but the average percentage of improvement was 1.91% in favor of our model. NPL collection shows the highest improvement of 11.13%, having 0.1819 for the vector model and 0.2048 for our model. Regarding normalized precision, the average percentage of improvement was 0.39% for CISI, 0.18% for CACM, and 0.21% for NPL. We preformed a statistical test to assess the significance of results (sign test) to check whether the results indicate that our model indeed improves precision. The null hypothesis tested was that the vector model performs at least as well as our enhanced model. This hypothesis was rejected with p-value $< \alpha = 0.06$ for CACM and p-value $< \alpha = 0.05$ for NPL in terms of MAP measure. The hypothesis was also rejected for CISI with a p-value $< \alpha = 0.06$, in terms of normalized precision (NPREC).

5 Conclusions and Future Works

In this article, we have presented a proposal for representing documents and queries with terms and relations that, according to the experiments, has shown itself to be feasible, and able to encode noun phrases. The work in progress focuses on extracting other relations among terms, and using them to enrich the document representation. We plan to encode several relations, enriching the vector representation one at a time. The relations we are planning to add are: name entities, subject-verb, verb-object, adjective-noun, and adverb-verb. A suitable weighting scheme for these new relations has to be defined. Later on, larger collections will be indexed and used for retrieval experiments. It seems reasonable to conjecture, based on our results, that this new representation and retrieval model will allow obtaining higher precision, when compared to the classical vector model. In contrast to the work in [9] that identifies composed terms and adds them to the classic vector representation, in this proposal, the representation is enriched in order to obtain benefits, not only in retrieval information, but also in other information processing tasks, such as question answering and classification. To illustrate this, assume that we want to answer: Who was Pilates? After identifying the named en-

tity in this query, Pilates surely will be a person named entity. Therefore, the query vector will have the encoded relation: *per* ⊗ Pilates, where *per* represents a special pattern similar to *left* in section 3. If we have the following paragraphs:

1. "The Pilates method also develops in those who practice skills such as attention and discipline. In addition…"
2. "A German named Pilates, born in the late nineteenth century, had a childhood full of health problems. Asthma, rheumatic fever, rickets. Calamities …"

The vector for paragraph 1 will be built as the addition of its term vectors: pilates + method + develops +… Meanwhile, paragraph 2 will have a vector like: german+ name +(*per* ⊗ Pilates)+ born+… So, looking for the encoded relation *per*⊗ Pilates of the query, paragraph 2 will be ranked higher than 1, leading to the answer. We are working on tagging named entities in the collections, so they can be extracted, represented and used for retrieval, and later for question answering.

Acknowledgments: The first author was supported by scholarship 217251/208265 granted by Conacyt, while second author was partially supported by SNI, Mexico.

References

1. Becker J., Kuropa D.: Topic-based Vector Space Model. In: Procs. of the 6th International Conference on Business Information Systems, pp. 7-13, July 2003 Colorado, USA.
2. Evans D., Zhai C.: Noun-phrase Analysis in Unrestricted Text for Information Retrieval. In: Procs. of the 34th Annual Meeting on Association for Computational Linguistics, pp. 17-24, June 1996.
3. Gonçalves A., Zhu J., Song D., Uren V., Pacheco R.: LRD: Latent Relation Discovery for Vector Space Expansion and Information Retrieval. In: Procs. of the Seventh International Conference on Web-Age Information Management, pp. 122-133, June 2006, Hong Kong, China.
4. Grinberg D., Lafferty J. and Sleator D.: A Robust Parsing Algorithm for Link Grammars. Carnegie Mellon University, Computer Science, Technical Report CMU-CS-95-125, 17p.,1995.
5. Lewis D., Sparck K.: Natural Language Processing for Information Retrieval. In: Communications ACM 39, pp. 92-101, January 1996.
6. Mitra M., Buckley C., Singhal A., Cardie C.: An Analysis of Statistical and Syntactic Phrases. In: Procs. of RIAO-97, 5th International Conference, pp. 200-214.
7. Plate T.A.: Analogy Retrieval and Processing with Distributed Vector Representation, Victoria University of Wellington, Computer Science, Technical Report CS-TR-98-4, 16 p.
8. Shi S., Wen J., Yu Q., Ruihua R., Ying Ma W.: Gravitation-Based Model for Information Retrieval. In: Procs. of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval 2005, pp. 488-495, Salvador, Brazil August 15 - 19, 2005.
9. Vilares J., Gómez-Rodríguez C. and Alonso M.A.: Managing Syntactic Variation in Text Retrieval. In: Peter R. King, Procs. of the 2005 ACM Symposium on Document Engineering. Bristol, United Kingdom, pp. 162-164, November 2-4, 2005, ACM Press, New York, USA.

Efficient Two-Phase Data Reasoning for Description Logics

Zsolt Zombori

Abstract Description Logics are used more and more frequently for knowledge representation, creating an increasing demand for efficient automated DL reasoning. However, the existing implementations are inefficient in the presence of large amounts of data. We present an algorithm to transform DL axioms to a set of function-free clauses of first-order logic which can be used for efficient, query oriented data reasoning. The described method has been implemented in a module of the DLog reasoner openly available on SourceForge to download.

Introduction

Description Logics (DL) consitute a family of languages designed for conveniently describing domain specific knowledge of various applications. The existing implementations for automated DL reasoning are mostly based on the so called tableau method which works just fine deducing new rules from existing ones, but it is rather slow when it comes to dealing with large amounts of data. In practice, however, the latter situation is becoming more and more typical.

We have developed the DLog system, an efficient DL data reasoner. This program can handle a data quantity that is too much to be loaded into main memory and hence can only be accessed through direct database queries. The reasoning task is broken into two parts: in the first phase only the rules of the knowledge base are considered and the second phase constitutes the data reasoning. The present paper deals with the first phase.

Section 1 gives a summary of description logics and first-order resolution, as well as a resolution based solution for DL theorem proving. Section 2 constitutes the core of this paper: it presents the first phase of DLog, i.e., how to transform the

Zsolt Zombori
Department of Computer Science and Information Theory, Budapest University of Technology and Economics, e-mail: zombori@cs.bme.hu

Please use the following format when citing this chapter:

Zombori, Z., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 393402.

rules of a DL knowledge base into a function-free set of clauses which forms the basis of the subsequent efficient query driven data reasoning. Section 3 gives a brief overview to the DLog system which is described in detail in [5].

1 Background

This section gives a recollection of some notions necessary to understand the paper and gives references to relevant sources.

1.1 Description Logics

Description Logics (DLs) [3] is family of logic languages designed to be a convenient means of knowledge representation. They can be embedded into FOL, but – contrary to the latter – they are decidable which gives them a great practical applicability. A DL knowledge base consists of two parts: the TBox (terminology box) and the ABox (assertion box). The TBox contains rules that hold in a specific domain. The ABox stores knowledge about individuals. This paper is concerned with a language called SHIQ which is a widespread DL language, thanks to a good compromise between complexity and expressivity.

We are interested in the following reasoning task: for a given SHIQ knowledge base KB and a query expression Q, we would like to decide whether Q is a logical consequence of KB. If Q contains no variables we expect a yes/no answer. If variables appear in the query, we would like to obtain the complete list of constants that, when substituted for the variables, result in assertions that follow from KB.

1.2 Resolution

Resolution [7] is a complete method for proving first order theorems. Its two inference rules are summarised in Figure 1 where σ is the most general unifier of B and C ($\sigma = MGU(B,C)$. *Ordered resolution* [2] refines this technique by imposing an

$$\frac{A \vee B \qquad \neg C \vee D}{A\sigma \vee D\sigma} \qquad\qquad \frac{A \vee B \vee C}{A\sigma \vee C\sigma}$$

Fig. 1 Binary Resolution and Positive Factoring

order in which the literals of a clause can be resolved. This reduces the search space while preserving completeness. It is parametrised with an *admissible ordering* ($\succ$) on literals and a *selection function*. *Basic superposition* [1] is an extension of ordered resolution with explicit rules for handling equality. The rules are summarised

Hyperresolution	$\frac{(C_1 \vee A_1) \ldots (C_n \vee A_n) \quad (D \vee \neg B_1 \vee \cdots \vee \neg B_n)}{(C_1 \vee \cdots \vee C_n \vee D)\sigma}$
Positive factoring	$\frac{A \vee B \vee C}{A\sigma \vee C\sigma}$
Equality factoring	$\frac{C \vee s = t \vee s' = t'}{(C \vee t \neq t' \vee s' = t')\sigma}$
Reflexivity resolution	$\frac{C \vee s \neq t}{C\sigma}$
Superposition	$\frac{(C \vee s = t) \quad (D \vee E)}{(C \vee D \vee E[t]_p)\sigma}$

Fig. 2 Inference rules of Basic Superposition

in Figure 2, where $E|_p$ is a subexpression of E in position p, $E[t]_p$ is the expression obtained by replacing $E|_p$ in E with t, C and D denote clauses, A and B denote literals without equality and E is an arbitrary literal. The necessary conditions for the applicability of each rule are given in the following list:

Hyperresolution: *(i)* σ is the most general unifier such that $A_i\sigma = B_i\sigma$, *(ii)* each $A_i\sigma$ is maximal in $C_i\sigma$, and there is no selected literal in $(C_i \vee A_i)\sigma$, *(iii)* either every $\neg B_i$ is selected, or $n = 1$ and nothing is selected and $\neg B_1\sigma$ is maximal in $D\sigma$.
Positive factoring: *(i)* $\sigma = MGU(A,B)$, *(ii)* $A\sigma$ is maximal in $C\sigma$ and nothing is selected in $A\sigma \vee B\sigma \vee C\sigma$.
Equality factoring: *(i)* $\sigma = MGU(s,s')$, *(ii)* $t\sigma \not\succ s\sigma$, *(iii)* $t'\sigma \not\succ s'\sigma$, *(iv)* $(s = t)\sigma$ is maximal in $(C \vee s' = t')\sigma$ and nothing is selected in $(C \vee s = t \vee s' = t')\sigma$.
Reflexivity resolution: *(i)* $\sigma = MGU(s,t)$, *(ii)* in $(C \vee s \neq t)\sigma$ either $(s \neq t)\sigma$ is selected or nothing is selected and $(s \neq t)\sigma$ is maximal in $C\sigma$.
Superposition: *(i)* $\sigma = MGU(s, E|_p)$, *(ii)* $t\sigma \not\succ s\sigma$, *(iii)* if $E = {}'w = v'$ and $E|_p$ is in w then $v\sigma \not\succ w\sigma$ and $(s\sigma = t\sigma) \not\succ (w\sigma = v\sigma)$, *(iv)* $(s = t)\sigma$ is maximal in $C\sigma$ and nothing is selected in $(C \vee s = t)\sigma$, *(v)* in $(D \vee E)\sigma$ either $E\sigma$ is selected or nothing is selected and $E\sigma$ is maximal, *(vi)* $E|_p$ is not a variable position.

An important feature of basic superposition is that it remains complete even if we disallow superposition into variables or terms substituted for variables. Such positions are referred to as *variable positions* or *marked positions* and are surrounded with '[ ]'.

1.3 Resolution Based Reasoning for DL

In [6] a resolution based theorem proving algorithm for the SHIQ DL language is presented. The knowledge base, together with the query expression is transformed into a set of FOL clauses with a characteristic structure, called ALCHIQ clauses and are summarised in Figure 3, where:

- **P**(t) is a possibly empty disjunction $(\neg)P_1(t) \vee \cdots \vee (\neg)P_n(t)$ of unary literals;
- **P(f**(x)): is a possibly empty disjunction $\mathbf{P_1}(f_1(x)) \vee \cdots \vee \mathbf{P_n}(f_n(x))$;
- term t is not marked, [t] is marked and <t> may or may not be marked;
- $\# \in \{=, \neq\}$;

Fig. 3 ALCHIQ clauses

$$\neg R(x,y) \vee S(y,x) \tag{1}$$

$$\neg R(x,y) \vee S(x,y) \tag{2}$$

$$\mathbf{P}(x) \vee R(x, < f(x) >) \tag{3}$$

$$\mathbf{P}(x) \vee R([f(x)], x) \tag{4}$$

$$\mathbf{P_1}(x) \vee \mathbf{P_2}(< \mathbf{f}(x) >) \vee \bigvee (< f_i(x) > \# < f_j(x) >) \tag{5}$$

$$\mathbf{P_1}(x) \vee \mathbf{P_2}([g(x)]) \vee \mathbf{P_3}(< \mathbf{f}([g(x)]) >) \bigvee (< t_i > \# < t_j >) \tag{6}$$

where t_i and t_j are of the form $f([g(x)])$ or of the form x

$$\mathbf{P_1}(x) \vee \bigvee_{i=1}^{n} (\neg R(x, y_i)) \vee \bigvee_{i=1}^{n} \mathbf{P_2}(y_i) \vee \bigvee_{i,j=1}^{n \times n} (y_i = y_j) \tag{7}$$

$$\mathbf{R}(< a >, < b >) \vee \mathbf{P}(< t >) \vee \bigvee (< t_i > \# < t_j >) \tag{8}$$

where t, t_i and t_j are either a constant or a term $f_i([a])$

The reasoning task is reduced to deciding whether the obtained FOL clauses are satisfiable. This is answered using basic superposition extended with a method called *decomposition*. [6] shows that the set of ALCHIQ clauses is bounded and that any inference with premises taken from a subset N of ALCHIQ results in either *(i)* an ALCHIQ clause or *(ii)* a clause redundant in N[1] or *(iii)* a clause that can be decomposed to, i.e., substituted with two ALCHIQ clauses without affecting satisfiability. These results guarantee that the saturation of an ALCHIQ set terminates.

1.4 Separating TBox and ABox Reasoning

The drawback of the above resolution algorithm is that it can be painfully slow. Resolution with saturation is a bottom-up strategy and computes all logical consequences of the clause set, many of which are irrelevant to the current question. It would be nice to use some query oriented, top-down mechanism, however, such mechanisms are available only for more restrictive FOL languages, such as Horn Clauses. One can get around this problem by breaking the reasoning into two tasks:

[1] A redundant clause is a special case of other clauses in N and can be removed.

first perform a resolution based preprocessing to deduce whatever could not be deduced otherwise and then use a fast top-down reasoner.

Note that complex reasoning is required because of the rules (TBox) and that in a typical real life situation there is a small TBox and a large ABox. Furthermore, the rules in the TBox are likely to remain the same over time while the ABox data can change continuously. Hence we would like to bring forward all inferences involving the TBox only, perform them separately and then let the fast reasoner (whatever that will be) do the data related steps when a query arrives.

In the framework of basic superposition, when more than one inference steps are applicable, we are free to choose an order of execution, providing a means to achieve the desired separation. Elements from the ABox appear only in clauses of type (8). [6] gives two important results about the role of ABox axioms in the saturation process:

Theorem 1. *An inference from ALCHIQ clauses results in a conclusion of type* (8) *if and only if there is a premise of type* (8)*.*

Theorem 2. *A clause of type* (8) *cannot participate in an inference with a clause of type* (4) *or* (6)*.*

In light of Theorem 1, we can move forward ABox independent reasoning by first performing all inference steps involving only clauses of type (1) – (7). [6] calls this phase the saturation of the TBox. Afterwards, Theorem 2 allows us to eliminate clauses of type (4) and (6). This elimination is crucial because in the remaining clauses there can be no function symbol embedded into another. The importance of this result comes out in the second phase of the reasoning, because the available top down mechanisms are rather sensitive to the presence of function symbols.

By the end of the first phase DL reasoning has been reduced to deciding the satisfiability of FOL clauses of type (1) – (3), (5), (7) and (8), where every further inference involves at least one premise of type (8). For the second phase, [6] uses a datalog engine which requires function-free clauses. Therefore (unary) functional relations are transformed to new binary predicates and new constant names are added: for each constant a and each function f the new constant a_f is introduced to represent $f(a)$. Note that this transformation requires processing the whole ABox.

2 Towards Pure Two-Phase Reasoning

In this section we introduce modifications to the saturation of ALCHIQ clauses. We do this to be able to perform more inferences before accessing the ABox. This is not just a mere regrouping of tasks, we will see that the algorithm produces a crucially simpler input for the second phase with a huge impact on its performance efficiency and on the available data reasoning algorithms. The improvement is achieved by eliminating function symbols from the clauses derived from the TBox.

The initial SHIQ DL knowledge base was function-free. Then, after translating TBox axioms to FOL we eliminatee existential quantifiers using Skolemisation

which introduced new function symbols. The ABox remained function-free, hence everything that is to know about the functions is contained in the TBox. This means we should be able to perform all function-related reasoning before accessing the ABox.

2.1 The Modified Calculus

We modify basic superposition presented in 1.2 by altering the necessary conditions to apply each rule. The new conditions are given below, with the newly added conditions underlined:

HyperresolutionTBox: *(i)* σ is the most general unifier such that $A_i\sigma = B_i\sigma$, *(ii)* each $A_i\sigma$ is maximal in $C_i\sigma$, and either there is no selected literal in $(C_i \vee A_i)\sigma$ or A_i contains a function symbol, *(iii)* either every $\neg B_i$ is selected, or $n = 1$ and $\neg B_1\sigma$ is maximal in $D\sigma$, *(iv)* none of the premises contain constants.
HyperresolutionABox: *(i)* σ is the most general unifier such that $A_i\sigma = B_i\sigma$, *(ii)* each $A_i\sigma$ is maximal in $C_i\sigma$, and there is no selected literal in $(C_i \vee A_i)\sigma$, *(iii)* either every $\neg B_i$ is selected, or $n = 1$ and nothing is selected and $\neg B_1\sigma$ is maximal in $D\sigma$, *(iv)* each A_i is ground, *(v)* $D\sigma$ is function-free.
Positive factoring: *(i)* $\sigma = MGU(A,B)$, *(ii)* $A\sigma$ is maximal in $C\sigma$ and either nothing is selected in $A\sigma \vee B\sigma \vee C\sigma$ or A contains a function symbol.
Equality factoring: *(i)* $\sigma = MGU(s,s')$, *(ii)* $t\sigma \not\succ s\sigma$, *(iii)* $t'\sigma \not\succ s'\sigma$, *(iv)* $(s = t)\sigma$ is maximal in $(C \vee s' = t')\sigma$ and either nothing is selected in $C\sigma$ or $s = t \vee s' = t'$ contains a function symbol.
Reflexivity resolution: *(i)* $\sigma = MGU(s,t)$, *(ii)* in $(C \vee s \neq t)\sigma$ either $(s \neq t)\sigma$ is selected or $s \neq t$ contains a function symbol or nothing is selected and $(s \neq t)\sigma$ is maximal in $C\sigma$.
Superposition: *(i)* $\sigma = MGU(s, E|_p)$, *(ii)* $t\sigma \not\succ s\sigma$, *(iii)* if $E = {}'w = v'$ and $E|_p = w|_{p'}$ then $v\sigma \not\succ w\sigma$ and $(s\sigma = t\sigma) \not\succ (w\sigma = v\sigma)$, *(iv)* $(s = t)\sigma$ is maximal in $C\sigma$ and either nothing is selected in $(C \vee s = t)\sigma$ or $s = t$ contains a function symbol, *(v)* in $(D \vee E)\sigma$ either $E\sigma$ is selected or nothing is selected and $E\sigma$ is maximal, *(vi)* $E|_p$ is not a variable position.

Note that hyperresolution is broken into two rules (HyperrresolutionTBox and HyperresolutionABox) which differ only in the necessary conditions. In the following by *original calculus* we refer to the basic superposition presented in Section 1.2 and by *modified calculus* we mean the rules of basic superposition with the restrictions listed above. We will prove that the new calculus can be used to solve the reasoning task.

Proposition 1. *The modified calculus remains correct and complete.*

Proof. The inference rules of basic superposition are all valid even if we do not impose any restrictions on their applicability. Since in the new calculus only the conditions are altered, it remains correct.

The modifications that weaken the requirements to apply a rule only extend the deducible set of clauses, so they do not affect completeness.

In case of hyperresolution, let us first consider only the new condition *(iv)* and disregard condition *(v)* on HyperresolutionABox. The original hyperresolution step has a main premise of type (7) and of the side premises some are of type (3) – (4) and some of type (8). This can be broken into two by first resolving the main premise with all side premises of type (3) and (4) (HyperresolutionTBox) and then resolving the rest of selected literals with side premises of type (8) (HyperresolutionABox). A hyperresolution step in the original calculus can be replaced by two steps in the modified one, so completeness is preserved.

We now turn to condition *(v)* on HyperresolutionABox. Let us consider a refutation in the original calculus that uses a hyperresolution step. If all side premises are of type (3) and (4) then it can be substituted with a HyperresolutionTBox step. Similarly, if all side premises are of type (8), then we can change it to HyperresolutionABox, as clauses of type (7) are function-free, satisfying condition *(v)*. The only other option is that there are both some premises of type (3) and of type (8)[2]. The result of such step is a clause of the following type:

$$\mathbf{P_1}(x) \vee \bigvee \mathbf{P_2}(a_i) \vee \bigvee \mathbf{P_2}([f_i(x)]) \vee$$
$$\vee \bigvee (a_i = a_j) \vee \bigvee ([f_i(x)] = [f_j(x)]) \vee \bigvee ([f_i(x)] = a_j)$$

At some point each function symbol is eliminated from the clause (by the time we reach the empty clause everything gets eliminated). In the modified calculus we will be able to build an equivalent refutation by altering the order of the inference steps: we first apply HyperresolutionTBox which introduces all the function symbols, but none of the constants, then we bring forward the inference steps that eliminate function symbols and finally we apply HyperresolutionABox. The intermediary steps between HyperresolutionTBox and HyperresolutionABox are made possible by the weakening of the corresponding necessary conditions. Notice, that by the time HyperresolutionABox is applied, functions are eliminated so condition *(v)* is satisfied.

We conclude that for any proof tree in the original calculus we can construct a proof tree in the modified calculus, so the latter is complete. □

Proposition 2. *Saturation of a set of ALCHIQ clauses with the modified calculus terminates.*

Proof. (sketch) We build on the results in [6], that clauses of type (8) are initially of the form $C(a), R(a,b), \neg S(a,b), a = b$ or $a \neq b$, i.e., they do not contain any function symbols. We will also use the fact that in the original calculus any inference with premises taken from a subset N of ALCHIQ results in either *(i)* an ALCHIQ clause or *(ii)* a clause redundant in N or *(iii)* a clause that can be substituted with two ALCHIQ clauses via decomposition.

All modifications (apart from breaking hyperresolution into two) affect clauses having both function symbols and selected literals, in that we can resolve with the

[2] It is shown in [6] that clauses of type (8) and (4) participating in an inference result in a redundant clause so we need not consider this case.

literal containing the function symbol before eliminating all selected literals. Such a clause can only arise as a descendant of a HyperresolutionTBox step. After applying HyperresolutionTBox, we obtain a clause of the following form:

$$\mathbf{P_1}(x) \vee \bigvee(\neg R(x,y_i)) \vee \bigvee \mathbf{P_2}(y_i) \vee \bigvee \mathbf{P_2}([f_i(x)]) \vee \\ \vee \bigvee(y_i = y_j) \vee \bigvee([f_i(x)] = [f_j(x)]) \vee \bigvee([f_i(x)] = y_j) \tag{9}$$

In the following, it will be comfortable for us to consider a clause set that is somewhat broader than (9), in which function symbols can appear in inequalities as well. This set is:

$$\mathbf{P_1}(x) \vee \bigvee(\neg R(x,y_i)) \vee \bigvee \mathbf{P_2}(y_i) \vee \bigvee \mathbf{P_2}([f_i(x)]) \vee \\ \vee \bigvee(y_i = y_j) \vee \bigvee(< f_i(x) > \# < f_j(x) >) \vee \bigvee(< f_i(x) > \# \, y_j) \tag{10}$$

where $\# \in \{=, \neq\}$. Of course, every clause of type (9) is of type (10) as well.

Let us see what kind of inferences can involve clauses of type (10). First, it can be an superposition with a clause of type (3) or (5). In the case of (3) the conclusion is decomposed (in terms of [6]) into clauses of type (3) and (10), while in the case of (5) we obtain a clause of type (10). Second, we can resolve clauses of type (10) with clauses of type (10) or (5). The conclusion is of type (10). Finally, we can apply HyperresolutionABox with some side premises of the form $R(a,b_i)$, but notice that only if the literals with function symbols are missing. The result is of type (8). This means that during saturation, we will only produce clauses of type (1) – (8) and (10). It is easy to see that there can only be a limited number of clauses of type (10) over a finite signature[3]. Hence the modified calculus will only generate clauses from a finite set, so the saturation will terminate. □

2.2 Implementing Two-Phase Reasoning

We use the modified calculus to solve the reasoning task in two phases. Our separation differs from that of [6] in that function symbols are eliminated during the first phase, without any recourse to the ABox. The method is summarized in Algorithm 1, where steps (1) – (3) constitute the first phase of the reasoning and step (4) is the second phase, i.e., the data reasoning.

Proposition 3. *A function-free ground clause can only be resolved with function-free clauses. Furthermore, the conclusion is ground and function-free.*

Proof. It follows simply from the fact that a constant a cannot be unified with a term $f(x)$ and from condition *(v)* on HyperresolutionABox. □

We are now ready to state our main claim:

[3] We already know from [6] that the set of clauses of type (1) – (8) is finite.

Algorithm 1 SHIQ reasoning

1. We transform the SHIQ knowledge base to a set of clauses of types (1) - (8), where clauses of type (8) are function-free.
2. We saturate the TBox clauses (types (1) - (7)) with the modified calculus. The obtained clauses are of type (1) - (7) and (10).
3. We eliminate all clauses containing function symbols.
4. We add the ABox clauses (type (8)) and saturate the set.

Proposition 4. *Algorithm 1 is a correct, complete and finite DL theorem prover.*

Proof. We know from Proposition 2 that saturation with the modified calculus terminates. After saturating the TBox, every further inference will have at least one premise of type (8), because the conclusions inferred after this point are of type (8) (Proposition 3). From this follows, (using Proposition 3) that clauses with function symbols will not participate in any further steps, hence they can be removed. In light of this and taking into account that the modified calculus is correct and complete (Proposition 1), so is Algorithm 1. □

2.3 Benefits of Eliminating Functions

The following list gives some advantages of eliminating function symbols before accessing the ABox.

1. It is more **efficient**. Whatever ABox independent reasoning we perform after having accessed the data will have to be repeated for every possible substitution of variables.
2. It is **safer**. A top-down reasoner dealing with function symbols is very prone to fall into infinite loops. Special attention needs to be paid to ensure the reasoner doesn't generate goals with ever increasing number of function symbols.
3. ABox reasoning without functions is **qualitatively easier**. Some algorithms, such as those for datalog reasoning, are not available in the presence of function symbols. We have seen in Section 1.4 that [6] solves this problem by syntactically eliminating functions, but this requires scanning through the whole ABox, which might not be feasible when we have a lot of data.

3 The DLog System

The DLog system is a complete SHIQ DL reasoner which incorporates the results presented in this paper. As an input it takes a SHIQ knowledge base and the TBox is first transformed to a set of function-free clauses based on [6] and Section 2.1. The resulting clauses are next used to build a Prolog program. It is the execution of this

program – run with an adequate query – that performs the data reasoning. The transformation to Prolog uses the PTTP approach, a complete theorem prover technology for FOL [8]. The readers interested in the DLog system should consult [5]. The program is also available at `http://dlog-reasoner.sourceforge.net`.

We compared the performance of DLog with three description logic reasoning engines: RacerPro 1.9.0, Pellet 1.5.0 and the latest version of KAON2. KAON2 implements the methods described in [6] and hence it is in many ways similar to DLog. For a thorough performance evaluation see [5]. Here we only mention that the larger the ABox, the better DLog performed compared to its peers. Also, to our best knowledge, DLog is the only DL reasoner which doesn't need to scan through the whole ABox and load it to main memory, enabling it to reason over really large amounts of data stored in external databases.

Summary

This paper showed how to extend the results in [6] to transform a SHIQ TBox into a set of first-order clauses. The particularity of these clauses is that they have a rather simple structure, namely they are function-free. This opens the way for fast query oriented inference algorithms to perform data reasoning tasks originally formulated in DL. The DLog system illustrates how this can be achieved, though it should be noted that the transformation presented here doesn't use anything specific to the DLog and is available to other reasoning engines as well.

References

1. Bachmair, L., Ganzinger, H.: Strict basic superposition. Lecture Notes in Computer Science **1421**, 160–174 (1998). URL citeseer.ist.psu.edu/bachmair98strict.html
2. Bachmair, L., Ganzinger, H.: Resolution theorem proving. In: Handbook of Automated Reasoning, pp. 19–99 (2001). URL citeseer.ist.psu.edu/bachmair01resolution.html
3. Horrocks, I.: Reasoning with expressive description logics: Theory and practice. In: Proc. of the 18th Int. Conf. on Automated Deduction (CADE 2002), 2392, pp. 1–15. Springer (2002)
4. Horrocks, I., Kutz, O., Sattler, U.: The Even More Irresistible SROIQ. In: P. Doherty, J. Mylopoulos, C.A. Welty (eds.) KR, pp. 57–67. AAAI Press (2006). URL http://dblp.uni-trier.de/db/conf/kr/kr2006.html#HorrocksKS06
5. Lukácsy, G., Szeredi, P.: Efficient description logic reasoning in Prolog: the DLog system. Tech. rep., Budapest University of Technology and Economics (2008). URL http://sintagma.szit.bme.hu/lukacsy/publikaciok/dlog_tplp_submission.pdf. Submitted to Theory and Practice of Logic Programming
6. Motik, B.: Reasoning in Description Logics using Resolution and Deductive Databases. Ph.D. thesis, Univesität Karlsruhe (TH), Karlsruhe, Germany (2006)
7. Robinson, J.A.: A machine-oriented logic based on the resolution principle. J. ACM **12**(1), 23–41 (1965). DOI http://doi.acm.org/10.1145/321250.321253
8. Stickel, M.E.: A Prolog Technology Theorem Prover: A New Exposition and Implementation in Prolog. Theor. Comput. Sci. **104**(1), 109–128 (1992)

Some Issues in Personalization of Intelligent Systems: An Activity Theory Approach for Meta Ontology Development

Daniel E. O'Leary[1]

Abstract. Personalization of systems has been part of the renaissance of artificial intelligence in many domains. This paper investigates some emerging issues in the area of personalization as they impact systems from different perspectives. Particular attention is given to the relationship between explicitly and implicitly gathered information, information gathered from other personalization settings and with the generation of a personalization information ontology, based on an activity theory approach. Finally, some privacy issues are considered, potentially limiting information sharing between applications.

1 Introduction

Personalization has been part of the renaissance of artificial intelligence [1]. There has been substantial research on personalization. For example, personalization of interaction with hardware and software has been occurring in multiple settings such as, in web navigation [2,3], Internet commerce [4] and ambient intelligence [5]. In addition, personalization is being used and proposed in many industries, e.g., consumer electronics [5] and health care [6].

1.1 What is personalization?

Personalization has many characteristics, of which some or all may be employed in any particular setting, including the following. First, personalization tries to limit a user's workload and provide context by facilitating *remembering* key aspects of system use by a particular individual. At one extreme, this means providing a history of what has been examined, and when (e.g., web pages, files, etc.). Second, personalization can facilitate *security and privacy* by ensuring that only a particular user makes use of an application [7]. If the system is personalized

[1] University of Southern California, 3660 Trousdale Parkway, Los Angeles, CA 91011, USA email: oleary@usc.edu

Please use the following format when citing this chapter:

O'Leary, D.E., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 403412.

"enough" it would recognize an intruder. Third, personalization can be used to try and make applications more easy-to-use by presenting results in particular ways, ranging from font size to icon to language presentation to other issues. Fourth, personalization will adapt and evolve as the user's interest's change.

Personalization uses whatever data may be available, that is generated by particular individuals. Specific applications typically gather whatever data they can that will facilitate their need to provide personalization. In order to personalize, that data is then mined in order to generate information to facilitate modeling user behavior and interaction. Generally, personalization is based on what the user does, not what they say they do. Accordingly, data is gathered implicitly, in the background. However, in many settings, systems explicitly gather data regarding what the user wants or say that they want. Seldom do systems interact to share personalization data.

1.2 Purpose of this Paper

The purpose of this paper is to review some emerging issues in personalization systems. In particular, this paper is concerned with generating and sharing personalization information between different applications. In addition, it is concerned about what information is needed from or about the user in order to personalize particular functions and activities. Specifically, it is aimed at developing a theory-based meta-ontology that can be used in development of an ontology for personalization systems.

1.3 Outline of this Paper

The paper proceeds in the following manner. Section 1 has introduced the paper, discussed its purpose and provided an outline of the paper. Section 2 provides a brief review of the previous research. Section 3 investigates the process of personalization. Section 4 provides a brief discussion of activity theory, a theory of behavior that is used here as a basis for personalization. Section 5 uses activity theory as a basis for generating a meta-ontology that could be used to generate an ontology for personalization. Section 6 investigates the importance of information about time. Finally, Section 7 briefly summarizes the paper, discusses an extension and reviews some of the paper's contributions.

2 A Brief Review of Previous Research

In a short paper like this, the extent to which the previous literature can be reviewed is limited. However, there has been substantial previous research in many dimensions regarding the use of personalization. Jeevan and Padhi [8] offer a recent survey of the literature of content personalization, focusing on providing a bibliography of research in the area of personalization. Approaches to capturing

personalization have included using data mining [3], intelligent agents [4], ontologies [9], and other approaches.

Although there has been some investigations of ontologies for personalization, the primary focus of those efforts seems to have been in the generation of ontologies for individuals to facilitate search. However, there has been limited research investigating ontologies for personalization, in general.

3 The Process of Personalization: Data, and Mining the Data

The process of personalization requires that the system responsible for personalization gather data that allows the system to personalize itself or allows the user to personalize the system. This is done by explicitly gathering data from the user in the foreground, or implicitly gathering data in the background or from other sources that also may be gathering personal information.

3.1 Foreground Information Gathering

The system can gather information directly from the user as part of a configuration process in order to facilitate the personalization process. As an example, multilingual applications typically ask the user which language the user would like to use. Gathering personalization information in the foreground generally is seen as obtrusive. On the other hand, oftentimes, such data gathering is expected by the user or implementer to facilitate implementation. Data gathered in the foreground can be viewed as data that the user *says is true* or characteristic. Ultimately, such data may contrast with data gathered from the user as to what they actually do.

3.2 Background Information Gathering

The system also can gather information in the background, as the user works or otherwise makes use of the system or its interface.

As an example, [9] developed a system that "watches" a user and gathers information regarding a user's search while developing profiles based on concept hierarchies designed to facilitate future search for the user. Such background gathered information is based on what the user *actually* does.

3.3 Gather Personalization Information from Other Sources

It appears that in most settings, systems function independently of other systems, each gathering personalization information. However, since personalization is used in a number of settings and applications, one approach would be to gather personalization information from other applications, minimizing redundant work.

As a result, in these settings it would facilitate personalization if there were a standard set of processes and a standard language relating to personalization information, so that such information could be shared. Alternatively, if personalization data from one source could be captured and translated to another

source, that could also facilitate cross use of personalization information. For example, armed with an ontology of personalization information from one source, it would be possible to cross link that ontology with other ontologies to facilitate integration of both and cross use of personalization information.

There are some potential advantages and disadvantages of using personalization information from other sources. Advantages include less work and less time before a system is personalized. For example, if a system is able to directly import personalization information, then the user will not need to replicate, either explicitly or implicitly, generation of that information. This would speed personalization substantially. Disadvantages include misuse and misunderstanding of the information. Data gathered from other applications may not be as reliable as data gathered by the current application. Unfortunately, there is no guarantee that the data has been gathered in the particular setting etc. Further, there is no guarantee as to when the data was gathered, so the personalization could be dated. As a result, there may be concern over the source of the data. Finally, there may be privacy issues if applications share data. Some applications may be based on a web server or by other means provide personalization information to other sources.

3.4 Comparing What We Do and What We Say We Do

Personalization can allow us to map out differences between what we say and what we do, if both sets of data are captured, analyzed and compared.

If both data sets are available then we can compare the two, and in so doing provide a management capability. For example, data could be gathered with respect to when a particular assignment is due, and that information could be used to help a user make sure that they meet the specific deadline. The key issue is "are we doing what we said we wanted to do?" In this setting, the system becomes our alter ego, there to remind and, potentially cajole us to complete our tasks in a timely manner.

Second, if both sets of information are available, then they can be used to provide a quality check. If the two do not reconcile then that can indicate a difference that may deserve additional context generation.

Third, comparing the two data may provide a security check. In particular, if the two are not in sync, then the actual user may not be a legitimate user [7].

3.5 Forgetting Can be More Important

True personalization also requires knowing when things should be "remembered" and when things should be "forgotten." People change and their interests change. People forget things and move on to other activities. Accordingly, systems also need to forget. A system that is personalized to include old behaviors is not likely to be effective or personal.

4 Activity Theory

Research on human behavior has resulted in construction of what is referred to as "activity theory" [10,11,12]. Detailed discussion of activity theory is beyond the scope of this particular paper. However, because activity theory provides a model of human behavior in context, it also provides a basis for analyzing ontological requirements of personalization systems.

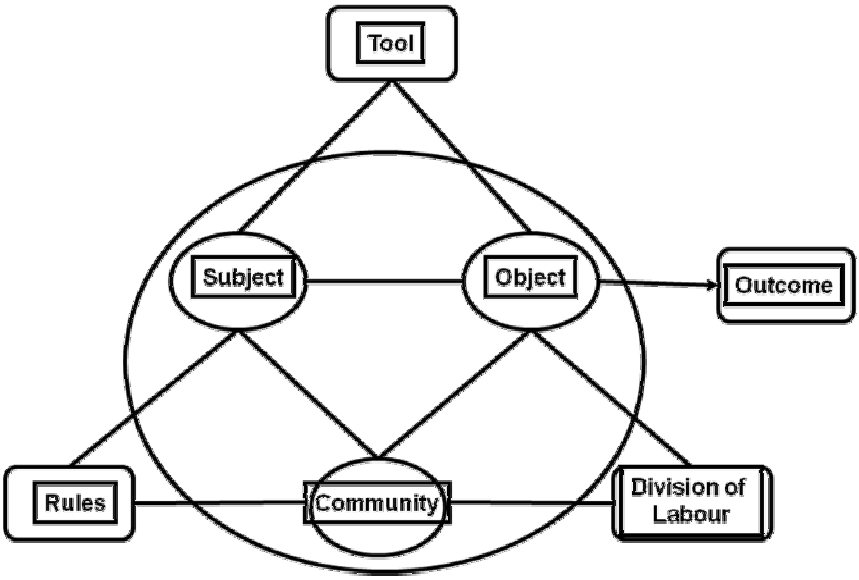


Figure 1 An Activity Theory Model of Behavior [12]

A summary of activity theory is provided in figure 1. Activity theory is based on the notion that subjects (people) perform "activities" or events in a context, using tools (e.g., software and hardware), rules (e.g., rules of interaction or behavior or organization rules), division of labour (each person has their own job to do), with an object of the activity (e.g., something being modified or created by the activity, such as a resource), while based in a community of others. Ultimately, the activity results in an outcome. As an example, in ecommerce, the activity could be a purchase of a pair of jeans, while in knowledge management it could be to produce knowledge artifacts. The subject would be the person making the purchase, the rules could be explicit or implicit (all my friends wear Levi's jeans, so I need to buy Levi's). The community could be the reference group, while the tools could be the computer, browser etc. Personalization would need to consider these broad categories, with respect to a particular individual.

5 Personalizing User Information Needs: Selected Ontology Requirements

Ontology requirements have been build to facilitate personalized search (e.g., [9, 13]) however, there has been only limited research structuring an ontology for personalization systems. Accordingly, this section outlines ontological requirements for personalization. In particular, using activity theory, we can anticipate certain characteristics that systems likely need to know ontology-based information about

- Personalization needs more than just history – *Activity* Types
- Personalization needs information about others – *Subject* Information
- Personalization systems need to consider the subject's position in their organization, i.e., the overall *community* in which the activity takes place.
- Some user focused objects are more "persistent" – *Object* Types
- Personalization occurs in a context where there may or may not be related available information – *Tool* information
- Personalization must take into account *rules* that the subject must follow
- Personalization needs to account for the particular portion of the project the subject is responsible for – *Division of labour*

5.1 Personalization Needs More than Just History – Activity Types

Some user information needs rapidly start and stop, after particular activities or events occur. Users have specific purposes and needs and when those purposes and needs are met there may not be a need to review any aspect of that process.

Consider a consumer who needs a new pair of jeans, so the outcome of the activity is the purchase of a new pair of jeans. After the jeans have been purchased, there is no longer a need for additional new jeans. For most people that means that they will not be searching for jeans for a long time. As a result, any history regarding the purchase of those jeans is not needed for a long time. After this purchase event there is limited interest in the history of this activity.

As another example, consider a researcher who is interested in examining other research papers on a topic, to first see what has been done in an area, and second, to see if a specific issue has been addressed, with the purpose of doing research on that topic and writing a paper on that topic. For most researchers, this means that once those papers have been identified, they are not interested in constantly reviewing the existing papers or finding new partially related papers. Recent history in these cases is not relevant to unfolding events.

In both of these cases, history is no longer important after some activity has occurred. Accordingly, the system needs to know that with the occurrence of some activities, there is no longer immediate interest in a particular topic. Histories must selectively remember and forget.

5.2 Subject Roles and Goals – Subject Information

The notion that different agents and participants have different roles long has been a part of system architectures. The nature of those roles is likely to vary based on the particular application. For example, [14] used basic economic roles of supply and demand to develop an architecture for intelligent agents in an ecommerce setting. Accordingly, roles and goals are likely to depend on the particular personalization application. However, in general, in personalization settings, a key agent role is one where the agent is required to "spy" on the user in the background, while another agent role may be to analyze data. A third role is likely to be one that makes sure that any appropriate rules, organizational or other are adhered to, as discussed below.

5.3 Some User Information Needs are Persistent, But Others Stop on Fulfillment – Object Types

In ecommerce and other settings, personalization systems need to take into account the type of goods, or objects being pursued. Shopping for stables, such as food, beer and wine and music is persistent over time. Where I have been in the past is helpful in the future.

However, other needs such as that new home loan or information regarding houses disappear as soon as I get that loan or buy that new house. The information needs for all goods are not persistent over time.

As a result, knowledge of the "persistence" of information needs is critical to a user's information needs about particular resources. Accordingly, if a system is to truly personalize for information needs over time, then there is a need to "understand" which information resource needs are persistent and which are not persistent.

5.4 Community Integration

Generally, personalization focuses on the individual, and does not consider the community in which the subject is based. Such issues can be critical. For example, if the subject is part of an organization, that organization needs to be considered. In addition, the behavior may be heavily influenced by reference groups. If so, perhaps personalization needs can be anticipated by understanding personalization needs of those in the reference groups.

5.5 Tools

Generally, personalization is limited to the software and hardware that the subject has access to. A complete ontology about personalization would need to consider the tools available to the user. Since computer-based tools evolve at such a rapid rate, an ontology of tools also would need the ability to evolve to accommodate such changes.

5.6 Rules

Subjects face a broad base of rules with which they must take into account. For example, in the case of organizations, there generally are rules that limit the amount that any one individual can purchase at any one point in time. These rules need to be accounted for as part of personalization, whether they are limiting the individual or all individuals in the group of which the subject is a member.

5.7 Division of Labour

Subjects often only perform a portion of some activity, as the tasks to complete an activity are assembled. As a result, personalization needs to take into account how labor is divided. Here an ontology would account for different jobs that need to be accommodated. Accordingly, the ontology could employ organization models, or models of different jobs.

5.8 Summary

The review of these different categories of information that map into the personalization context, suggests that no single ontology will meet all of these needs. For example, rules are likely to vary substantially from setting to setting, making it difficult for a single ontology to meet the needs of all personalization needs.

6 Some Information Needs are Time Dependent

Ontological needs also extend to concerns about time. As a result, there have been a number of ontologies developed for time including [15]. Such an ontology is important since personalization needs are dependent on what time it is, whether time of day, or time of year. For example,

- I am a soccer coach for my children. I am more concerned about soccer sites roughly from August through the beginning of December, the span of the soccer season in California.
- We have purchased toys on the Internet, usually in October or November.

How can systems get such time dependent information? There are at least two sources: user actions and user plans.

6.1 Time Dependent - User Actions

As we discussed earlier, data about user actions can be gathered unobtrusively focusing on what the user does at what point in time, and monitoring the time lines. For some settings, such as the examples listed above, multiple years of data would be required before a system could infer such results.

6.2 Time Dependent - User Plans

Gathering planning information for a single application can be difficult, because subjects have multiple simultaneous activities. So how could personalization systems find out about these alternative activities with minimal additional user activity? One approach would be to tie into a calendaring system, and the events and activities inherent in such settings[16]. This would require consistency between any calendaring ontology and any personalization ontology.

7 Summary, Extension and Contributions

This paper has investigated some issues of personalization of systems. Data for such systems can be gathered in the foreground or background. In addition, personalization can be facilitated by integrating personalization information from other systems. However, in order for systems to be able to talk with each other generally requires use of the same ontology or the ability to translate from one ontology to another.

Accordingly, this paper investigated the need for an ontology for personalization systems and laid out an initial outline of a meta ontology, based on activity theory, that could be used to generate such an ontology. Based on activity theory, the requirements for that ontology include knowledge of subjects, objects, community, rules, tools, division of labour and how time relates each.

7.1 Extensions

At some firms, such as 1-800-Flowers, there is a lot of emphasis on a determining and meeting user's specific needs as part of the information gathering process, such as sending flowers for a birthday or anniversary. In particular, the user's needs ultimately are linked to a transaction processing system. The transaction processing system behind generating the order gathers "occasion" data unobtrusively. That data can then be used to come back to the user and remind them of the occasions, and their "need" to send flowers. However, use of this information requires at least two developments. First, there needs to be a "link" between the existing order processing system that gathers the data and the personalization system. Information gathered as part of the transaction processing needs to used in the personalization system, rather than trying to re-gather the same or related data. Second, the ontologies used by the two systems must be "equivalent," at least for the variables of direct concern, e.g., "event," "occasion" or "birthday." Without equivalence, the system link will be limited. Generating and maintaining consistent ontologies is not an easily solved problem.

Finally privacy issues may limit the ability to share personalization information between applications. Alternatively, not using the same ontology across

applications can limit some potential problems with privacy concerns, and facilitate privacy preservation.

7.2 Contributions

This paper has noted that using personalization information from other systems generating personalization information can speed and ease personalization. In addition, a meta ontology for generating personalization ontologies was developed. Activity theory can provide a basis for knowing what kind of detailed ontology information will be needed in personalization systems. It also provides us with the insight that it is unlikely that a single ontology for personalization can be generated, because of broad ranges of diverse sets of rules that individuals function under, organizational settings, rapidly changing tools and other issues.

References

1. O'Leary, D., The Internet, Intranets and the AI Renaissance, Computer, Jan 1997.
2. O'Leary, D., AI and Navigation on the Internet and Intranet, IEEE Expert, Apr 1996, pp. 8-10.
3. Mulvenna, M., Anand, S., Bucher, A., Personalization on the Net using Web Mining, Communications of the ACM, Aug 2000, 43(8), pp. 122-123.
4. Lee, W-P, Liu, C-H and Lu, C-C, Intelligent Agent-based Systems for Personalized Recommendations in Internet Commerce, Exp Sys with Appl, Vol 22, 2002, pp. 275-284.
5. Aarts, E., Ambient Intelligence: A Multimedia Perspective, IEEE Multimedia, Vol 11(1), Jan – Mar 2004, pp. 12 -19.
6. Riva, G., Ambient Intelligence In Health Care, CyberPsychology & Behavior," 6(3), 2003, pp. 295-300.
7. O'Leary, D., Intrusion-Detection Systems, J of Inf Sys, Vol 6, 1, Spring, 1992, pp. 63-74.
8. Jeevan, V.K.J., Padhi, P., A selective review of research in content personalization, *Library* Review, 2006, Vol 55 (9), pp. 556-586.
9. Pretschner, A. and Gauch, S., Ontology Based Personalized Search, *Proc. 11th IEEE Intl.* Conf. on Tools with Artificial Intelligence, pp. 391-398, Chicago, Nov 1999.
10. Engestrom, Y. "Learning by Expanding an Activity Theoretical Approach to Developmental Research," Accessed on Apr 9, 2008, Orientakonsultit, Helsinki, 1987. http://lchc.ucsd.edu/MCA/Paper/Engestrom/expanding/toc.htm,
11. Nardi. B.(Ed), 1996, Context and Consciousness: Activity Theory and Human Computer Interaction, MIT press, Harvard Mass.
12. Sierhuis, M. and W. J. Clancey, Modeling and Simulating Work Practice: A Method for Work Systems Design, IEEE Intelligent Systems, pp. 32-41, Sept – Oct 2002.
13. Trajkova, J. and Gauch, S., Improving Ontology-based User Profiles, 2004, http://www.ittc.ku.edu/keyconcept/publications/RIAO2004.pdf, accessed on April 9, 2008.
14. Brown, C., Gasser, L., O'Leary, D., and Sangster, A., AI on the WWW: Supply and Demand Agents, IEEE Expert, Aug 1995, pp. 50-55.
15. Hobbs, J. and Pan, F., An Ontology of Time for the Semantic Web, 2004, ACM, http://www.isi.edu/~hobbs/hobbs-pan-TALIP.pdf, Accessed on Apr 9, 2008.
16. Cost, R., et al., "ITalks: A Case Study in the Semantic Web and DAML +OIL," IEEE Intelligent Systems, Jan – Feb 2002, pp. 40-47.

SHORT PAPERS

Smart communications network management through a synthesis of distributed intelligence and information

J. K. Debenham, S. J. Simoff, J. R. Leaney, and V. Mirchandani

Abstract. Demands on communications networks to support bundled, interdependent communications services (data, voice, video) are increasing in complexity. Smart network management techniques are required to meet this demand. Such management techniques are envisioned to be based on two main technologies: (i) embedded intelligence; and (ii) up-to-the-millisecond delivery of performance information. This paper explores the idea of delivery of intelligent network management as a synthesis of distributed intelligence and information, obtained through information mining of network performance.

1 Introduction

Communications networks are increasingly becoming integrated with user applications including core business systems. In addition, the core business systems are starting to incorporate highly interactive information-rich Web 2.0 technologies such as different architectures of participation (e.g. weblogs, wikis), peer-to-peer technologies (e.g. Skype, iPhone), many-to-many media publishing platforms (e.g. audio/video podcasts, Web TV, RSS feeds), social bookmarking and other social software [1]. These trends are democratising the creation of value for businesses [2] and provide significant enhancement of our knowledge creation and scientific capabilities. Technologically these trends contribute to the

J. K. Debenham, J. R. Leaney, V. Mirchandani
University of Technology, Sydney, e-mail: {debenham, johnl, vinodm}@it.uts.edu.au

S. J. Simoff
University of Western Sydney, e-mail: s.simoff@uws.edu.au

Please use the following format when citing this chapter:

Debenham, J.K., Simoff, S.J., Leaney, J.R. and Mirchandani, V., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 415419.

development of the intelligent ambient habitat [see [3], p.150] of the knowledge civilisation age[1]. As a result, requirements, placed on communication networks are becoming increasingly complex and personalised. These lead to the development of requirements specifications for *bundles of interdependent services*, often with complex mutual demands and constraints. These services are to be delivered over networks that comprise an increasing variety of heterogeneous network domains each with their differing operational and/or administrative policies. An example of dealing with such bundled, interdependent communications network is the Alcatel-Lucent triple play[2] architecture [4], which is being *provisionally* deployed by several Service Providers. This paper explores the idea of delivery of intelligent network management in such networks as a synthesis of distributed intelligence and information, obtained through information mining of network performance.

2 Network Resource Management

Network resource management is concerned with the allocation/deallocation of resources to support services that a service provider has committed to a customer. Centralised management approaches cannot cope with increasing management scale, and, there is now a focus on decentralisation and delegation of management decision-making. Service differentiation is a key driver of decentralisation. Products are based on increasing service personalisation, putting more complexity into provisioning processes. Services may be tailored to individual needs, or customers may be provided with seamless access to services across multiple devices or networks without reconfiguration [5]. Furthermore, context- and location-based services allow services to be tailored according to the user's location or interests. Therefore, the trend is moving towards more frequent and complex management activities, with per-customer or per-subscriber management. This trend is already evident in existing and next-generation broadband architectures, where per-subscriber policies are required [6]. Service differentiation is also driving more dynamic and adaptive management systems. The network needs to be reconfigured, depending on the user's environment and the available network resources. This is especially the case with respect to business-critical services that demand "five nines" availability (i.e. 99.999% which equates to only 5 minutes of downtime a year). Timely, adaptive and flexible management is necessary for dynamically changing network environments, for the introduction of new services on-demand [7], as well to evolve to accommodate changing user and Service Provider requirements.

---

[1] The term "knowledge civilization age is borrowed from 3. Wierzbicki, A.P., Nakamori, Y.: Creative Space: Models of Creative Processes for the Knowledge Civilization Age. Springer, Berlin/Heidelberg (2006)

[2] "Triple play" refers to voice, video and data services over an Internet Protocol (IP) based network.

Autonomic computing, which requires systems to be more self-managing, adaptive and aware of their environment demonstrates some potential to address network resource management issues, but there will need to be several enhancements – greater communication and negotiation, improved trustworthiness, visibility and accountability, and evolved intelligence [8].

A simpler approach is Management by Delegation (MbD), where management intelligence is shifted closer to the managed systems through Service Level Agreements (SLAs). These protocols are used for automated service negotiation. They include: (i) the dynamic service negotiation protocol (DSNP) for service level negotiation using a client-server architecture, and service level specification (SLS) negotiation at the IP layer; and (ii) the 'resource negotiation and pricing protocol' (RNAP) protocol allows the negotiation of prices for the contracted services. DSNP suffers from the drawbacks of a centralized architecture. RNAP has limited scalability because it relies on periodic signalling from subscribers to negotiated services. These and other protocols, presented in details in [9] contain useful ideas but are not capable of negotiating bundles of inter-dependent services from multiple service providers simultaneously.

3 Agent-based negotiation and Information mining in Communications Systems and Networks

The deployment of intelligent agents in communications systems has been widely investigated since the early-1990s. Sadly the potential has not yet been fully realised. What we believe to be the reason for this failure is also the central assumption of our approach: that the way to deliver intelligent network management is through a synthesis of distributed intelligence and as accurate as possible information about the performance of the network. Despite failing to deliver this vision, thirteen years later interest in agents in the communications industry continues to flourish. The eMarkets Group[1] has developed the QDINE[2] approach, which adopts a distributed open market approach to service management. Service charging is described within an SLA, allowing the use of any charging model appropriate to a provided service. The synthesis of information and decision-making has led the researchers from the group to merge information mining with intelligent agency in an on-going sequence of works on "Information-based Agency" [e.g. see [10, 11]]. The unifying theory underpinning this work is information theory. The deployment of this synthesis to communications systems is starting to gain interest [12]. Given the fundamental significance of information theory to communications, intelligent agents based on information theory seem to be the natural choice of technology for coupling

[1] http://research.it.uts.edu.au/emarkets

[2] The QDINE site is: http://qdine.it.uts.edu.au/. The public section of the site includes lists of the publications where one can find the details of the formal specification language for describing SLAs together with its ontology.

intelligence with data- and information mining in communications networks. Earlier work in the application of data mining in telecommunications has focused on fault isolation and forecasting telecommunication equipment failures based on mining communication network data, which describes the state of the hardware and software components in the network. These approaches, however, did not have embedded mechanisms to recognise and take in account various changes in communications networks and their predictive models are acceptable only over a limited period of time [13].

4 The Elements of the Smart Communications Network Management Technology

The elements of the proposed technological framework for smart communications network management that addresses the above discussed issues include:

- *Language and ontologies for formal representation of SLAs* about bundles of interdependent services from multiple service providers and an *SLA Negotiation Framework* that utilises these formal representations;
- *Algorithms for mining network predictive models and short-term network performance information*, which provide the necessary information about the trends and local variations in the network behaviour – this information can be used both during the negotiation of what is possible to deliver, i.e. the values of the different SLA parameters, and deliver what has been agreed, i.e. the strategies that enable the fulfilment of what the SLA promises.
- *Strategies for flexible resource provisioning* that enable bundling the 'best' mix of services based on the current data and flexible adapting of the service provider to the actual situation on the net in order to meet its obligation in the SLA;
- *Mechanisms that enable informed routing*, which rely on predictive models of network performance and by up-to-the-millisecond information correction before the actual routing of the packages is performed.

5 Conclusions

The synthesis of distributed intelligence and information, obtained through information mining of network performance has the potential to advance network resource management. Presented ideas address network performance management are looking at reducing the gap between current network performance, and, desired network performance, using artificial intelligence techniques which have been demonstrated to be flexible, and to scale. The challenge in this is to design a

system that operates quickly and does not place significant overhead on the network.

References

1. ISTAG: New Business Sectors in Information and Communication Technologies: The Content Sector as a case study. Information Society Technology Advisory Group (ISTAG) (2007)
2. Tapscott, D., Williams, A.D.: Wikinomics: How Mass Collaboration Changes Everything. Portfolio, London (2006)
3. Wierzbicki, A.P., Nakamori, Y.: Creative Space: Models of Creative Processes for the Knowledge Civilization Age. Springer, Berlin/Heidelberg (2006)
4. Kompella, V.: A Novel architecture for Triple Play services. Proceedings Asia Pacific Regional Internet Conference on Operational Technologies APRICOT 2005, Kyoto, Japan (2005)
5. Falcarin, P.: A CPL to Java compiler for dynamic service personalization in JAIN-SIP server. IEC Annual Review of Communications **57** (2004)
6. Anschutz, T.: DSL evolution - architecture requirements for the support of QoS-Enabled IP Services. DSL Forum Technical Report TR-059. Architecture and Transport Working Group (2003)
7. Sloman, M., Lupu, E.: Policy specification for Programmable Networks. Proceedings of the International Workshop on Active Networks, Berlin, Germany (1999)
8. Strassner, J.: Autonomic networking: Theory and practice. 9th IFIP/IEEE International Symposium on Integrated Network Management, IM05 (2005) 786 - 786
9. Sarangan, V., Chen, J.-C.: Comparative study of protocols for Dynamic Service Negotiation in the next-generation Internet. IEEE Communications Magazine (2006) 151-156
10. Zhang, D., Simoff, S.J., Debenham, J.K.: Exchange rate modelling for e-negotiators using text mining techniques. In: Lu, J., Ruan, D., Zhang, G. (eds.): E-Service Intelligence - Methodologies, Technologies and Applications. Springer, Heidelberg (2007) 191-211
11. Sierra, C., Debenham, J.: Information-Based Agency. Procedings of the 20th International Joint Conferences on Artificial Intelligence IJCAI07, Hyderabad, India (2007) 1513-1518
12. Rocha-Mier, L.E., Sheremetov, L., Batyrshin, I.: Intelligent agents for real time data mining in telecommunications networks. In: Gorodetsky, V., Zhang, C., Skormin, V.A., Cao, L. (eds.): Autonomous Intelligent Systems: Multi-Agents and Data Mining. Springer-Verlag (2007) 138-152
13. Weiss, G.M.: Data mining in the Telecommunications industry. In: Wang, J. (ed.): Encyclopedia of Data Warehousing and Mining. Information Science Publishing (2008)

An Abductive Multi-Agent System for Medical Services Coordination

Anna Ciampolini, Paola Mello, and Sergio Storari

Abstract We present MeSSyCo, a multi-agent system that integrates and coordinates heterogeneous medical services. Agents in MeSSyCo may perform different tasks such as diagnosis and intelligent resource allocation and coordinate themselves through an infrastructure based on a combination of abductive and probabilistic reasoning. In this way a set of specialized medical service providers could be aggregated into a system able to perform more complex medical tasks.

1 Introduction

Patients management usually needs complex and dynamic tasks since it requires the coordination of the services offered by several different and distributed medical organizations and resources (e.g., hospital departments, physicians, ambulances, etc.). The multi-agent paradigm seems to be the most appropriate approach to provide such features, and has been used in several works such as, e.g., in [7] and [9].

Following these considerations, this paper presents MeSSyCo, a multi-agent system whose main purpose is the coordination and integration of heterogeneous knowledge-based medical services. This system, recently proposed also for emergency scenario management [10], can be used to represent virtual organizations: each service provider is encapsulated into an agent; the coordination infrastructure allows the interaction of several (possibly heterogeneous) agents. Services may be implemented either by traditional programming technologies or by using knowledge-based systems with automatic reasoning mechanisms. In this way, the

Anna Ciampolini
DEIS, University of Bologna, 40100 viale Risorgimento 2 e-mail: aciampolini@deis.unibo.it

Paola Mello
DEIS, University of Bologna, 40100 viale Risorgimento 2 e-mail: pmello@deis.unibo.it

Sergio Storari
ENDIF, University of Ferrara, 44100 via Saragat 1 e-mail: sergio.storari@unife.it

Please use the following format when citing this chapter:

Ciampolini, A., Mello, P. and Storari, S., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 421426.

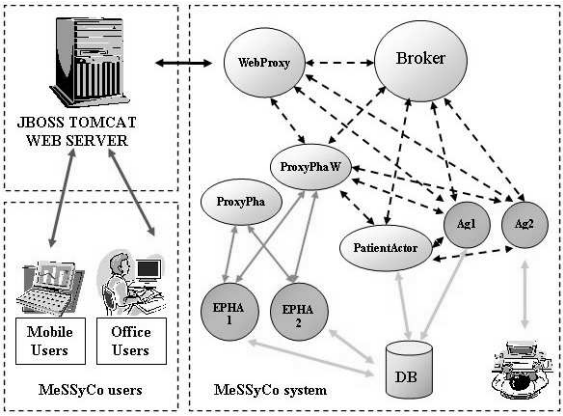


Fig. 1 Schema of the MeSSyCo system architecture.

MeSSyCo system is able to perform complex tasks such as intelligent resource allocation (identifying the most suitable resources), distributed service coordination (answering to complex service requests) and distributed diagnosis (combining heterogeneous knowledge to provide more precise diagnosis).

The MeSSyCo infrastructure is based on ALIAS [1], an extension of logic based abduction [3] to the multi-agent context. Abduction, a well known automatic hypothetical reasoning mechanisms that allows reasoning in presence of incomplete knowledge, is suitable for medical diagnosis since, given a set of symptoms, it can produce a set of plausible diagnosis for them.

In a real medical scenario, diagnosis should be further improved by considering probabilistic reasoning (see for example [6]). Merging such reasoning approach with the abductive one, make possible to associate a probability value to each plausible diagnosis and thus to identify the most realistic one.

To this purpose, in this paper we present a system that integrates the Probabilistic Horn Abduction (PHA) formalism [8], a well known approach particularly suited for medical diagnosis, with the coordination mechanisms provided by ALIAS.

2 The MeSSyCo Architecture

MeSSyCo can be considered a JADE [4] implementation of ALIAS [1] with some extensions regarding the distributed probabilistic reasoning and the identification of most appropriate service provider among the available ones. Its architecture, shown in Figure 1, is characterized by two kind of agents: the *application agents* and the *system agents*.

Each entity providing services within an organization is modeled by an application agent (shown in Figure 1 as *Ag1*, *Ag2*) which provides several services. Each application agent contains a reasoning module, described in Section 3, which stores

the knowledge used to provide each agent service. This knowledge may be elicited, for example, from clinician interviews or medical literature. It is also necessary to express how these entities interact with the others in order to accomplish their objectives.

System agents, shown in Figure 1, implement the services necessary to the correct functioning of the whole system. The *Broker* agent is an extension of the FIPA [2] *Directory Facilitator* agent, whose role is to identify, upon request, the most suitable agents that match with given requirements (specified in the request). Agents using the Distributed Probabilistic Horn Abduction (DPHA) reasoning methodology (described in Section 3), register their services into a dedicated broker named *ProxyPha*, that is the gateway between the non-DPHA agent and the DPHA agents. The *PatientActor* agent is a prototype agent able to retrieve information about a patient. The *WebProxy* agent allows secure access to MeSSyCo services.

3 Distributed Probabilistic Horn Abduction

In MeSSyCo we use a mix of abduction [3] and probabilistic reasoning for performing distributed diagnosis and selecting the "best" diagnosis among the set of plausible ones. The possibility of merging logical and probabilistic notions of evidential reasoning in a unifying computational framework based on abduction has been the subject of several works in literature.

A framework for merging abduction and probabilistic reasoning, has been proposed by Poole and named Probabilistic Horn abduction (PHA) [8]. This framework uses Horn-clauses with probabilities associated with hypotheses (abducibles) and incorporates assumptions about the rule base and independence assumptions among hypotheses. The language is that of pure Prolog with special disjoint declarations that specify a set of disjoint hypotheses with associated "a priori" probabilities. If Δ is the set of minimal explanations e_i of conjunction of atoms g from theory TH, we have that the probability of g is the sum of the probabilities of the e_i in Δ. If $\{h_1, ..., h_n\}$ are the hypotheses h_i in a minimal explanation e_i, then the probability of e_i is the product of the probabilities of the h_i in e_i. Poole showed how PHA can represent a discrete Bayesian network and how, given a set of evidences, it is possible to compute the "a posteriori" probability of the abducibles.

Starting from PHA and ALIAS, in MeSSyCo we defined the Distributed Probabilistic Horn Abduction (DPHA). The novelty is, with respect to Poole's work, that we coordinate several PHA agents, each enclosing its own knowledge base (KB). The goal of DPHA is to use these KBs in order to perform a probabilistic evaluation similar to the one achievable by a single agent with a complete KB. The agent KB contains: a set of rules describing relations among domain variables; a set of `disjoint` clauses describing the "a priori" probabilities of the abducibles and the probabilistic relations among domain variables.

The result of the execution of a DPHA service S is a set of N Plausible Set of Conclusions (PSC_k) $\{PSC_1, \ldots, PSC_k, \ldots, PSC_N\}$ where each PSC_k is expressed by

$([\,[C_{k1}, p(C_{k1})], \ldots, [C_{kM_k}, p(C_{kM_k})]\,],\ p(path_k),\ bunch_k)$, where C_{ki} is a conclusion (e.g. a pathology); $p(C_{ki})$ is the "a priori" probabilities associated to the C_{ki} conclusion; $p(path_k)$ is the probability associated to the reasoning path followed to obtain the PSC_k; $bunch_k$ is the set of agents who have collaborated to define PSC_k .

In the case of diagnosis, a conclusion (i.e., an *abducible*), represents a single pathology that may explain (possibly in combination with other pathologies) one or more symptoms. The probability associated with the query and with each PSC is computed in the same way proposed by Poole in PHA.

Suppose to have a Bayesian network which describes the relation among two abducibles, *Tuberculosis* (*tub*) and *Bronchitis* (*bro*), and one symptom, *Dyspnoea* (*dys*). This Bayesian network is represented in the DPHA KB of an agent *ag* as:

```
disjoint([ tub(y):0.4, tub(n):0.6 ]).
disjoint([ bro(y):0.3, bro(n):0.7 ]).
disjoint([c_dys(y,y,y):0.95, c_dys(n,y,y):0.05 ]).
disjoint([c_dys(y,y,n):0.85, c_dys(n,y,n):0.15 ]).
disjoint([c_dys(y,n,y):0.65, c_dys(n,n,y):0.35 ]).
disjoint([c_dys(y,n,n):0.05, c_dys(n,n,n):0.95 ]).
dys(Vd) <- tub(Vt),bro(Vb),c_dys(Vd,Vt, Vb)
```

If an agent asks *ag* to explain dyspnoea dis(y), *ag* provides a set of four PSCs. The first explanation is $\{(tub(y), 0.4), (bro(y), 0.3), (c\_dys(y,y,y), 0.95)\}$ and it is transformed in: $PSC_1 = ([\,[(tub(y), 0.4],\ [bro(y), 0.3]\,],\ 0.95,\ ag)$. The "a posteriori" probability of the $tub(y)$ can be computed subdividing the probabilities of the PSCs containing the $tub(y)$ abducibles with the sum of the probabilities of all the PSCs.

4 MeSSyCo Coordination Language

The MeSSyCo Coordination Language (MCL), derived from the coordination language described in [5], is used by agents to interact with other agents. As in ALIAS, coordination among agents is expressed in MCL using two composition operators: the collaborative operator # and the competitive operator (;). MCL language provides also a communication operator (>) that is used to submit queries to other agents. Competition is used when the same medical service can be provided by several agents whereas collaboration is used when a set of required services could not be provided by a single agent.

The query $A0 : A1 > (G1, ServiceSupplyType1, InitCond1, Abdin1)$ expresses that $A0$ asks $A1$ to solve $G1$, considering the prior knowledge $InitCond1$, the abducibles contained in $Abdin1$, the modality specified in $ServiceSupplyType1$; if $G1$ succeeds in $A1$, N ($N > 0$) Plausible Sets of Conclusions $PSC_{1i}(i \in [1, \ldots, N])$, consistent in the bunch $\{A0, A1\}$, could be obtained for $G1$.

A MCL collaborative query q formulated by $A0$ for service $G1$ provided by $A1$ and service $G2$ provided by $A2$ uses the collaborative operator # between the two distinct service requests. The result is a set of *PSC*s of the agent bunch $\{A0, A1, A2\}$, obtained computing the Cartesian product of the agent solutions. Each PSC_k is obtained making the union of the abducibles in PSC_{1i} and in PSC_{2i}: if they contains the same abducible but with a different value of the associated variable, for exam-

ple $con(y)$ and $con(n)$, PSC_k is inconsistent and deleted; if they contain the same abducible with the same value of the associated variable, in PSC_k we associate to it a probability that is the average of its probabilities in PSC_{1i} and PSC_{2i}. The probability associated to the reasoning path of PSC_k is obtained computing the product of the probability of the one of PSC_{1i} with the one of PSC_{2i}. The bunch of PSC_k is $\{A0,A1,A2\}$.

In the competitive query, $A0$ asks the service G to $A1$ and $A2$ by using the ; operator. The resulting set PSCq contains all the PSC for G, obtained joining both the PSC_{1i} of the bunch $\{A0,A1\}$ the PSC_{2j} of the bunch $\{A0,A2\}$. If both $A1$ and $A2$ fail, the competitive query fails.

5 Conclusion and Future Works

In this paper we focused on the definition and development of a multi-agent architecture for the management of heterogeneous medical services which uses abduction enriched with probabilistic notions to express agent reasoning in the case of diagnosis and also to manage the coordination between different agents. The resulting coordination framework, named Distributed Probabilistic Horn Abduction (DPHA), is able to join the results of distinct agent services into a unique abductive answer.

In the future, we plan to complete the MeSSyCo implementation, facing other important aspects related to the medical application field like data security and experiment it in real world scenario.

References

1. A. Ciampolini, E. Lamma, P. Mello and P. Torroni. An Implementation for Abductive Logic Agents. In *Proceedings AI*IA99*, Pitagora Editore, Bologna, Italy, (1999).
2. FIPA, see the web site: www.fipa.org, Accessed 14 March 2004.
3. A. C. Kakas and P. Mancarella. 'Generalized stable models: a semantics for abduction', In *Proc. 9th European Conference on Artificial Intelligence*, Pitman Pub, (1990).
4. JADE, see the web site: jade.cselt.it, Accessed 14 March 2004.
5. A. Ciampolini, P. Mello and S. Storari. 'Distributed Medical Diagnosis with Abductive Logic Agents', In *Proc. of ECAI2002 workshop on Agents applied in health care*, 23-32, Printed by the organizers, (2002).
6. P. Lucas, L. van der Gaag, A. Abu-Hanna. 'Bayesian Networks in Biomedicine and Health-Care', *Artificial Intelligence in Medicine*, **30**, (2004).
7. A. Moreno, D. Isern, D. Snchez. 'Provision of agent-based health care services', *AI Communications*, **16(3)**, 167-178, (2003).
8. D. L. Poole. 'Probabilistic Horn Abduction and Bayesian Networks', *Artificial Intelligence*, **64(1)**, 81-129, (1993).
9. B. Lopez, S. Aciar, B. Innocenti, I. Cuevas. 'How multi-agent systems support acute stroke emergency treatment', in *IJCAI Workshop on Agents Applied in Health Care*, 51-59, (2005).
10. A. Ciampolini, P. Mello, S. Storari. 'An abductive multi-agent framework for distributed service coordination and reasoning in emergency scenarios', *Workshop on MObile and DIstributed approaches in Emergency Scenarios*, IEEE press, (2008).

A New Learning Algorithm for Neural Networks with Integer Weights and Quantized Non-linear Activation Functions

Yan Yi[1], Zhang Hangping[2], and Zhou Bin[3]

Abstract The hardware implementation of neural networks is a fascinating area of research with for reaching applications. However, the real weights and non-linear activation function are not suited for hardware implementation. A new learning algorithm, which trains neural networks with integer weights and excludes derivatives from the training process, is presented in this paper. The performance of this procedure was evaluated by comparing to multi-threshold method and continuous discrete learning method on XOR and function approximation problems, and the simulation results show the new learning method outperforms the other two greatly in convergence and generalization.

1 Introduction

In recent years, Feedforward Neural Networks (FNNs) have been widely used in the areas of pattern recognition, signal processing, time series analysis, and many others. Most of these applications need to be implemented with monolithic integrated circuit or digital signal processor (DSP) in real word. However, the mapping of resultant networks onto fast, compact, and reliable hardware is a difficult task. The problem is that the conventional multilayer FNNs, which have continuous weights, are expensive to store weights and implement calculation in digital hardware. FNNs with integer weights are easier and less expensive to implement in electronics and the storage of the integer weights is much easier to be achieved. The training algorithm in this paper proposes an effective solution for hardware implementation of small-scale FNNs.

There have been some researches focusing on this area. A multiple-threshold method (MTM) has been proposed for generating discrete-weight FNNs

[1] Prof. Yan Yi
Institute of Intelligence &Software, Hangzhou Dianzi University, Hangzhou, China
email: yybjyyj@163.com
[2] Zhang Hangping
email: qingying1226@yahoo.com.cn
[3] Zhou Bin
email: zhoubin02051502@yahoo.com.cn

Please use the following format when citing this chapter:

Yi, Y., Hangping, Z. and Bin, Z., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 427431.

(CHIEUEH et al., 1988; WOODLAND, 1989). In this simple method, the continuous weights of a fully trained FNN are quantized into discrete valued weights using a nonlinear function (usually a multiple-threshold). The continuous discrete learning method (CDLM) (E. Fiesler et al., 1990) follows a more fruitful strategy. In this method, a trained continuous weight network is quantized. The errors obtained from the discrete network are back-propagated through the continuous network and then trained again. This cycle repeats until the network converges.

Unfortunately, all methods above are unable to discretize the non-linear activation function as a look-up table in training process since they are based on the BP algorithm which needs derivatives of the activation function.

Our main result is a new learning algorithm called optimum descent point learning method (ODPLM), which trains FNNs with integer weights and excluding derivatives of the activation function.

The remainder of this paper is divided into three sections. The first one proposes the new learning algorithm. The second section presents experiments and computer simulation results. The last section presents our conclusion and discussion.

2 Optimum Descent Point Learning Method

ODPLM falls under the category of performing learning, in which the network parameters are adjusted to optimize the performance of the network. The error function *E(X)* is always used to measure the performance of a network quantitatively, and the form of error function is

$$E(X) = \frac{1}{2}\sum_{p=1}^{P} E_p = \frac{1}{2}\sum_{p=1}^{P}\sum_{i=1}^{NL}(d_{pi} - y_{pi})^2 \qquad (1)$$

where X is the matrix of network weights and biases, E_p is the sum of the mean squares of errors associated with the pattern p, d_{pi} is the desired response of an output neuron at the input pattern p, y_{pi} is the response of an output neuron at the input patter p.

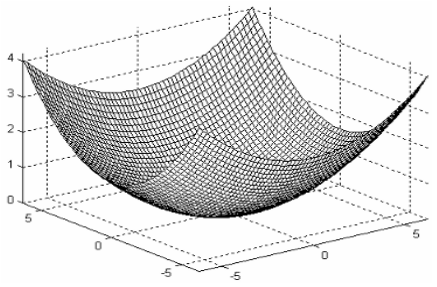


Figure. 1. A performance surface with minimal point (0,0)

The purpose of ODPLM is to search the point with the smallest error function in the parameter

space which has been discretized and confined to integers. The searching process is iterative. We begin from initial guess with integers, X_0, and then search the optimum neighbour which has the smallest error function among all neighbours as the next guess X_{k+1}. A neighbour of X_k is defined as

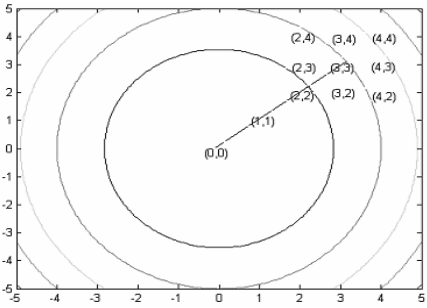


Figure. 2. The contour lines of fig. 1

$$X_k^{\,i} = X_k + p_i \qquad i \in 3^n \qquad (2)$$

where n is the size of matrix X_k, and p_i is a matrix composed of n elements in the set {-1, 0, 1}. All optimum points at each stage construct a path in the discretized space, along which the error function descends steepest.

The size of searching space is 3^n if we want to exhaust all combinatorial possibilities of n elements of p_i. Since the qualities of the final weights are focused on more than the efficiency of a method for off-line training, the exhaustive method is used to search the optimum neighbour for small-scale neural networks, and a further discussion for dealing with large-scale networks is presented in the fourth section.

Figure. 1 shows a performance surface with minimal point (0, 0), and fig. 2 shows the contour lines of fig. 1. When X_k is at the point (3, 3) in fig. 2, points (2, 4), (3, 4), (4, 4), (2, 3), (3, 3), (4, 3), (2, 2), (3, 2), (4, 2) are its neighbours. The point (2, 2) is selected as X_{k+1} since its error function is smallest among eight neighbours. Similarly, the point (1, 1) will be selected as the next optimum point after the point (2, 2), and the process continues until reaches at the minimum point (0, 0).

The activation function can be quantised as a look-up table in training process since OPDLM do not need derivatives. This eliminates the new inaccuracy resulted from the limited size of the look-up table when the fully trained network with integer weights is implemented by hardware.

```
Algorithm 1: <Optimum descent point learning method>
Step1:
      Quantise continuous weights space.
Step2:
      Quantise non-linear activation functions as a look-up
      table.
Step3:
       Intialize the FNNs with integer weights denoted by X_k.
While (e > E_allowed)
    Calculate the error functions of all neighbours of X_k
    with  exhaustive search
    choose the neighbour with the minimal error function as X_k+1
    if(E(X_k+1) > E(X_k))
```

```
        the process has been stuck in local minimum, therefore it
        should be terminated and make a fresh start with a new set
        of initial weights.
        Break
      End if
  End While
```

3 Functionality Tests

The classical learning test problem – the approximation of a sine curve function – has been used for testing the functionality. The reported parameters in the Tables for simulations that have reached solution are: *min* the minimum number of iterations, *mean* the mean number of iterations, *max* the maximum number of iterations, *time* the mean time of successful training processes, *succ*. simulations succeeded out of ten.

Table 1. Software simulation results on the approximation of the function f(x)

	MTM	CDLM	ODPLM
min	3185	8623	6
max	8461	23824	14
mean	6807	11694	11
time	4	20	420
succ.	0%	40%	100%

Let's assume that we want to approximate the following functions:

$$f(x) = e^{-x}\sin(2\pi x),$$

and the training set is obtained by sampling the function at the points x = 0, 0.1, 0.2, …, 0.9, 1. (There are total of 11 input/target pairs.) To approximate this function we will use a 1-4-1 network, where the activation function for the first layer is log-sigmoid and the activation function for the second layer is linear. The allowed error function is 0.01; the range of initial weights is (-10, 10); the learning rate is 0.1.

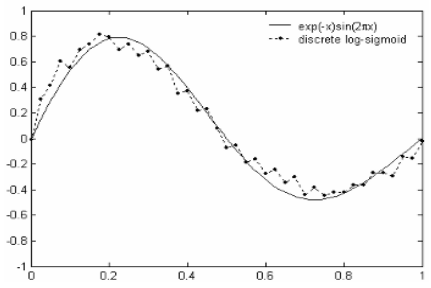


Figure. 3. Generalization of the network trained by ODPLM on the approximation of f(x)

The convergence performance of MTM, CDLM, and ODPLM on function approximation problems is shown in table 1. From the tables, we can see that ODPLM outperforms MTM and CDLM greatly in the successful training number out of ten. In addition, the epochs of ODPLM for each process are far less than

the other two. However, ODPLM requires a long computational time for each epoch, so the total running time is longer than the others.

Figure 3 presents the generalization of the best networks trained by ODPLM, in the figure the point-dotted line represents the responses of neural networks with discrete sigmoid function which has been quantised as a 50-size look-up table.

4 Conclusion and Discussion

A new learning algorithm ODPLM is presented in this paper, which trains the FNNs with integer weights and quantized activation functions. In the algorithm, non-linear activation functions have been already quantized in training process, therefore the inaccuracy will not increase in hardware implementation. The simulation results show the new learning algorithm works better than the CDLM and the multi-threshold method in terms of convergence and generalization.

References

1. A.H. Khan and E.L. Hines (1994) Integer-weight neural nets. ELECTRONICS LETTERS, 21st July, vol.30 No.15
2. CHIEUEH, T.D., and GOODMAN, R.M. (1988) Learning algorithms for neural networks with ternary weights. First Annual Meeting of INNS. Boston, MA. Pages 166.
3. E. Fiesler, A. Choudry, H.J. Caulfield. (1990). A weight discretization paradigm for optical neural networks. Proceedings of the International Congress on Optical Science and Engineering. Bellingham, Washington, U.S.A.: The International Society for Optical Engineering Proceedings, volume SPIE-1281, pages 164-173.
4. Martin T. Hagan, Howard B. Demuth(1996) Neural Network Design, PWS Publishing Company
5. MARCHESI. M., BENVENUTO, N., ORLANDI, G., PIAZZI. F., and UNCINI (1990), A. Design of multi-layer neural networks with power-of two weights. IEEE ISCS, New Orleans, pp. 2951-2954
6. RUMELHART, D.E., HINTON, G.E., and WILLIAMS, R.J. (1986) Learning internal representation by error backpropagation. Parallel distributed processing: Explorations in the microstructure of cognition. pp. 318-362.
7. T. Lundin, E. Fiesler and P. Moerland (1996) Connectionist Quantization Functions. The Proceedings of the 1996 SIPAR-Workshop on Parallel and Distributed Computing Geneve, Switzerland.
8. V.P. Plagianakos, M.N. Vrahatis. (2000). Training Neural Networks with Threshold Activation Functions and constrained Integer Weights. Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference. Volume 5, pages 161 -166.
9. VON LEHMEN, A., PAEK, E.G., LIAO, P.F., MARRAKCHI,.A, and PATEL, J.s. (1988). Factors influencing learning by backpropagation. Proc.Int. Conf. Neural Networks, San Diego, USA, pp. 335-341.

Neural Recognition of Minerals

Mauricio Solar, Patricio Perez, and Francisco Watkins

Abstract: The design of a neural network is presented for the recognition of six kinds of minerals (chalcopyrite, chalcosine, covelline, bornite, pyrite, and energite) and to determine the percentage of these minerals from a digitized image of a rock sample. The input to the neural network corresponds to the histogram of the region of interest selected by the user from the image that it is desired to recognize, which is processed by the neural network, identifying one of the six minerals learned. The network's training process took place with 160 regions of interest selected from digitized photographs of mineral samples. The recognition of the different types of minerals in the samples was tested with 240 photographs that were not used in the network's training. The results showed that 97% of the images used to train the network were recognized correctly in the percentage mode. Of the new images, the network was capable of recognizing correctly 91% of the samples.

1. Introduction

Chile is privileged in terms of rich mineral resources, particularly copper ores. One of the most important activities carried out by copper mining companies is prospecting for ores. Mining prospection requires a substantial amount of economic resources to determine the feasibility of going into a large investment to operate a mine. Prospecting consists in sampling rocks from different areas and determining the ore grade existing in those lands. To determine the grade of an ore the composition of the minerals present in the samples must be studied.

The procedure used to recognize minerals takes place by getting information on the different minerals that make up a rock. From these rocks obtained from the land that is being prospected, polished samples a quarter of an inch thick are prepared. Polishing of the samples provides smooth surfaces for analysis under a mi-

Mauricio Solar
Univ. Técnica Federico Santa María, Av. Sta María 6400, Santiago, Chile,
Univ. of Santiago de Chile, Av. Ecuador 3659, Santiago, Chile,
e-mail: msolar@inf.utfsm.cl
Patricio Perez
Univ. of Santiago de Chile, Av. Ecuador 3659, Santiago, Chile,
e-mail: pperez@usach.cl
Francisco Watkins
Univ. of Santiago de Chile, Av. Ecuador 3659, Santiago, Chile,
e-mail: watkins@usach.cl

Please use the following format when citing this chapter:

Solar, M., Perez, P. and Watkins, F., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and practice II*; Max Bramer; (Boston: Springer), pp. 433437.

croscope. The polished sample, which is called a **briquette,** is placed over a grid as a way to discretize the image under the microscope. In this way an expert quantifies the number of points covered or occupied by each *grain* of each mineral existing in the briquette to determine the percentage of that mineral in the sample. A grain is a section of the sample that contains a given mineral, and it is identified by the expert by referring to the visual characteristics of each mineral.

Both the process of recognition as well as the determination of the percentage is extremely slow and error-prone because of the excessive dependence on the attention of the expert in charge of the detection and counting of the points. An expert takes about 15 minutes to determine the total percentage of each mineral in a briquette.

In the literature there are applications for the automatic identification of minerals by different techniques. In [1] a digital image processing and texture analysis technique is shown for recognizing six kinds of rocks, with results showing 89% of correct recognition of 58 photographs. In [2] there is an automatic classification of the shape of graphite particles in cast iron. In [3] traditional multivariate statistical methods and extensions to address the problem of classifying minerals common in siliciclastic and carbonate rocks are applied. Other techniques like genetic programming have been developed in [4] to recognize mineral grains.

The classical algorithms are a viable alternative [5], however considering the associative processing implied in the use of neural engineering leads to fast and reliable automatic results. On the other hand, it contributes a practical application alternative to this important field of ore prospecting in Chile.

Section 2 details the problem and the design of the neural network (NN) to solve it; the process and the algorithms implemented are presented in section 3; section 4 shows the results obtained; and finally the conclusions are presented.

2. Analysis of the Problem and the Proposed Solution

In the first place it must be stated that the problem put forth is divided into two relevant areas: a) Recognition of the mineral; and b) counting points in the area associated with each grain, with the purpose of determining its percentage presence in the sample.

In order to design the NN's architecture to identify the patterns associated with each mineral, the learning process was started by training the NN with the previously digitized input patterns. The final product is a system that is capable of recognizing the mineral by inspection of the digitized image of a briquette. The input consists in the selection by the user of a region of interest (ROI) in the briquette, and the output is the identification of the mineral. Another alternative of the system is to select the *Percentages* option, which makes possible an exhaustive inspection of the whole image to determine the total amount of the mineral in the image (as a percentage).

It should be pointed out that to determine the copper grade, according to the experts it is enough to consider the presence of six minerals that contain the copper: chalcopyrite, chalcosine, covelline, bornite, pyrite and energite. For that reason the NN was trained to recognize the patterns of those six minerals.

Under constant data capture conditions it can be stated that one of the 6 minerals studied in a digitized sample has similar texture in different samples. Based on this, the NN's input is a **histogram** of the ROI of the digitized image that corresponds to the counting of the number of pixels classified according to its color level. The NN's architecture must be capable of learning these textures by classifying their characteristic histograms. After analyzing the characteristics of the 6 minerals that contain copper, it was considered possible to compress the histogram to 23 intensities, which are sufficient to distinguish the minerals.

The learning process took place through a backpropagation NN, based on the histogram of a ROI of 15x15 pixels of the image, and identifying that ROI with a given mineral. After determining the number of neurons in the hidden layer using the process indicated in [6], the NN had the following characteristics:

- 23 input units corresponding to the intensities of the histogram;
- a 13-unit hidden layer; and
- a 7-unit output layer, one to identify each mineral and another one for an unrecognized mineral.

The state of an output neuron *j* in layer *s* is given by Eq. 1. The input histogram is mapped in the [0,1] interval, which corresponds to the highest sensitivity region of the transfer function *f(z)*.

$$x_j^{[s]} = f\left(\sum_i \left(w_{ji}^{[s]} x_i^{[s-1]}\right)\right), \quad \text{where } f(z) = \left(1+e^{-z}\right)^{-1} \tag{1}$$

$w_{ji}^{[s]}$ is the connection weight from neuron *i* in layer *s*-1 to neuron *j* in layer *s*. The weights are fitted iteratively as shown in Eq. 2.

$$\Delta w_{ji}^{[s]} = \varepsilon^{[s]} \varepsilon_j^{[s]} x_i^{[s-1]} + \alpha^{[s]} \Delta w_{ji}^{[s]}(t-1) \tag{2}$$

$\varepsilon_j^{[s]}$ measure of local error in neuron *j* of layer *s*, $\varepsilon^{[s]}$ learning coefficient of layer *s* and $\alpha^{[s]}$ momentum coefficient.

To determine the number of neurons in the hidden layer, successive trainings were carried out gradually increasing the number of neurons in the hidden layer, according to the process indicated in [6]. The best results after this fit were obtained with 13 neurons in the hidden layer, with the following parameters: $\varepsilon^{[1]} = 0.4$; $\varepsilon^{[2]} = 0.4$; $\alpha^{[1]} = 0.8$; $\alpha^{[2]} = 0.8$; number of training cycles: 60,000; and the learning coefficient was decreased 10% every 10,000 cycles.

3. Mineral Recognition Process per Rectangle

In the recognition process a method of selecting the ROI was implemented in which the expert selects a rectangle of variable size of the image using the mouse, and then applies the procedure detailed below.

a. The histogram of the selected ROI is generated with 256 values.
b. The 256 values are transformed into an interval of 22 (interval 23 is set at 0), to generate the input to the NN.
c. The size of the selected ROI is normalized at the standard size through which the NN learned (15x15 pixels). This normalization is linear, leaving the result in a vector with 23 values which is passing by for the NN.
d. Every value of this vector is normalized again because it must be in the [0,1] range to be able to go into the backpropagation NN.
e. Finally, the input vector to the NN is available and it is processed by means of the propagation algorithm, getting the result in the output vector.
f. Six of the 7 output units represent a mineral. If there is an excited neuron above the value 0.6 (determined by the expert), then it is highly probable that the selected ROI is that mineral. If no neuron reaches the threshold, the sample is not sufficiently clear, and the background neuron is excited (Table 1).

Table 1. Minerals for determining the copper grade

Output	Mineral
oi[0]	Chalcopyrite
oi[1]	Chalcosine
oi[2]	Covelline
oi[3]	Bornite
oi[4]	Pyrite
oi[5]	Energite
oi[6]	Background

The recognition process by percentage carries out an exhaustive coverage of the image, scanning it totally through two cycles in which it considers the size of the ROI selected by the expert (for this case it was *5*). The result is found as the amount of i^{th} mineral was recognized.

4. Results

The NN implemented was trained with 160 digitized photographs of samples obtained directly from the prospected land. Those 160 photographs analyzed by the expert allowed the NN to be trained. To evaluate the recognition of the different types of minerals in the samples, a test was made with 240 photographs that were not used in the training of the NN. The results showed that 97% of the images

used to train the NN were recognized correctly in the percentage mode. Of the new images submitted to the NN, it was capable of recognizing correctly 91% of the samples.

The problems of poor classification can be attributed to the fact that some ROI of the images show superposition of two or more minerals, making the histograms unclear. This problem can be solved using a smaller window.

Every mineral sample in which the expert must determine the percentage of the minerals takes about 15 minutes per briquette. The photographs of the 400 briquettes mean 100 hours of work, and considering 8 hours per workday, they require 12.5 days from the expert. The automatic recognition of the 400 photographs takes less than 20 minutes, which is a substantial improvement in the time used for this process of recognition.

5. Conclusions

In the experts' opinion, the results obtained indicate that the type of neural network described here allows a satisfactory automation of the process of mineral recognition for the problem of prospecting for copper ores.

In the mineral recognition mode the system is simple to use. It is only required to select the ROI that it is desired to recognize, and the system indicates the degree of certainty of the recognized mineral.

In the percentage mode, the automatic system described showed to be reliable in a high percentage of correct recognition (93%) and fast when compared with the time taken by an expert for that work.

References

1. Wang, L., 1995, Automatic Identification of Rocks in Thin Sections Using Texture Analysis. *Mathematical Geology*. v. 27, no. 7, p. 847-865.
2. Gomes, O., Paciornik, S., 2003, Automatic Classification of the Shape of Graphite Particles in Cast Iron, *Microscopy and Microanalysis*, v. 9, p. 756-757.
3. Flesche, H., Nielsen, A.A., Larsen, R., 2000, Supervised Mineral Classification with Semiautomatic Training and Validation Set Generation in Scanning Electron Microscope Energy Dispersive Spectroscopy Images of Thin Sections, *Mathematical Geology*, v. 32, no. 3, p. 337-366.
4. Ross, B.J., Fueten, F., Yashkir, D.Y., 2001, Automatic Mineral Identification Using Genetic Programming, *Machine Vision and Applications*. v. 13, no. 2, p. 61-69.
5. Wang, W., Lei, L., 2006, Pattern Recognition and Computer Vision for Mineral Froth, *ICPR 2006, 18th In. Conf. on Pattern Recognition*, v. 4, p. 622-625.
6. Foresee, D., Hagan, M., 1997, Gauss-Newton Approximation to Bayesian Learning, In: *Int Joint Conf. on Neural Networks*, p. 1930-1935.

Bayesian Networks Optimization Based on Induction Learning Techniques

Paola Britos [1], Pablo Felgaer [2], and Ramon Garcia-Martinez [3]

Abstract Obtaining a bayesian network from data is a learning process that is divided in two steps: structural learning and parametric learning. In this paper, we define an automatic learning method that optimizes the bayesian networks applied to classification, using a hybrid method of learning that combines the advantages of the induction techniques of the decision trees with those of the bayesian networks.

1 Introduction

Data mining tasks can be classified in two categories: descriptive data mining and predictive data mining; some of the most common techniques of data mining are the decision trees (TDIDT), the production rules and neuronal networks. On the other hand, an important aspect in the inductive learning, is to obtain the dependency data between the variables involved in the phenomenon, in the systems where it is desired to predict the behavior of some unknown variables based on certain known variables, a representation of the knowledge that is able to capture this information on the dependencies between the variables is the bayesian networks [1]. A bayesian network is a directed acyclic graph in which each node represents a variable and each arc a probabilistic dependency, in which specifies the conditional probability of each variable given its parents; the variable at which it points the arc is dependent (cause-effect) of the variable in the origin of this one.

[1] Paola Britos

PhD Program, Computer Science School, La Plata University. CAPIS-ITBA. pbritos@itba.edu.ar

[2] Pablo Felgaer

Intelligent Systems Laboratory. School of Engineering. University of Buenos Aires. pfelgaer@fi.uba.ar

[3] Ramon Garcia-Martinez

Software & Knowledge Engineering Center (CAPIS), ITBA. rgm@itba.edu.ar

Please use the following format when citing this chapter:

Britos, P., Felgaer, P. and Garcia-Martinez, R., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 439443.

Obtaining a bayesian network from data is a learning process that is divided in two phases: the structural learning and the parametric learning. First of them, consists of obtaining the structure of the bayesian network, that means, the relations of dependency and independence between the involved variables. The second phase has the purpose to obtain the a priori and conditional probabilities from a given structure. Some characteristics of the bayesian networks are that they allow to learn dependency and causality relations, they allow to combine knowledge with data [2] and they can handle incomplete data [1] [3]. The bayesian networks can make the classification task -a particular case of prediction- that it is characterized to have a single variable of the database (class) that is desired to predict, whereas all the others are the data evidence of the case that is desired to classify. A great amount of variables in the database can exist; some of them directly related to the class variable but also other variables that have not direct influence on the class. In this work, a method of automatic learning is defined that helps in the pre-selection of variables, optimizing the configuration of the bayesian networks in classification problems.

2 Proposed hybrid learning method

We propose a hybrid learning method that combines the advantages of the induction decision trees techniques with those of the bayesian networks. For it, we integrate to the process of structural and parametric learning of the bayesian networks, a previous process of pre-selection of variables. In this process, it is chosen from all the variables of the domain, a subgroup with the purpose of generating the bayesian network for the particular task of classification and this way, optimizing the performance and improving the predictive capacity of the network. The method for structural learning of bayesian networks is based on the algorithm developed by Chow and Liu to approximate a probability distribution by a product of probabilities of second order, which corresponds to a tree. The joint probability of variables can be represented like:

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i) P(X_i | X_{j(i)}) \quad (1)$$

where Xj(i) is the cause or parent of Xi. Consider the problem like one of optimization and it is desired to obtain the structure of the tree that comes near more to the “real” distribution. A measurement of the difference of information between the real distribution (P) and the approximate one (P*) is used:

$$I(P, P^*) = \sum_x P(X) \log(P(X) / P^*(X)) \quad (2)$$

Then the objective is to minimize I. A function based on the mutual information between pairs of variables is defined as:

$$I(X_i, X_j) = \sum_x P(X_i, X_j) \log(P(X_i, X_j) / P(X_i) P(X_j)) \quad (3)$$

In this context, to find the more similar tree is equivalent to find the tree with greater weight. Based on that, the algorithm to determine the optimal bayesian network from data is shown on table 1.

Table 1. Algorithm to determine the optimal bayesian network

```
1. Calculate the mutual information between all the pairs of variables (n(n-1)/2).
2. Sort the mutual information in descendent order.
3. Select the arc of greater value as the initial tree.
4. Add the next arc while it does not form cycles. If it is thus, reject.
5. Repeat (4) until all the variables are included (n - 1 arcs).
```

Rebane and Pearl (1989) extended the algorithm of Chow and Liu for poly-trees. In this case, the joint probability is:

$$P(X) = \prod_{i=1}^{n} P(X_i \left| X_{j1(i)}, X_{j2(i)}, ..., X_{jm(i)} \right.) \quad (4)$$

where {Xj1(i), Xj2(i),…, Xjm(i)} is the set of parents for the variable Xi. In order to compare the results obtained when applying the complete bayesian networks (RB-Complete) and the preprocessed bayesian networks with induction algorithms C4.5 (RB-C4.5), we used the databases "Cancer" and "Cardiology" obtained at the Irving Repository of Machine Learning databases of the University of California [4]. Table 2 summarizes these databases in terms of amount of cases, classes, variables (excluding the classes), as well as the amount of resulting variables of the preprocessing with the induction algorithm C4.5.

Table 2. Databases description

Database	Variables	Variables C4.5	Classes	Control cases	Validation cases	Total cases
Cancer	9	6	2	500	199	699
Cardiology	6	4	2	64	31	95

The algorithm used to carry out the experiments with each one of the evaluated databases, is detailed in table 3. The step (1) of the algorithm makes reference to the division of the database in the control and the validation ones. In most cases, the databases obtained from the mentioned repositories were already divided. For the pre-selection of variables by the induction algorithms C4.5 of the step (2), we introduced each one of the control databases in a decision trees TDIDT generating system. From there, we obtained the decision trees that represent each one of the analyzed domains. The variables that integrate this representation perform the subgroup that was considered for the learning of the preprocessed bayesian networks. Next (3) a ten iteration process begins, in each one of these iterations processed 10%, 20%, 100% of the control database for the networks structural and parametric learning. The objective of the repetitive structure of the step (3.1) is to minimize the accidental results that do not correspond with the reality of the model in study.

Table 3. Algorithm used to carry out the experiments

```
1. Divide the database in two. One of control or training (approximately 2/3 of
   the total database) and the another one of validation (with the remaining
   data)
2. Process the control database with the induction algorithm C4.5 to obtain the
   subgroup of variables that will conform the RB-C4.5
3. Repeat for 10%, 20%, …, 100% of the control database
   3.1. Repeat 30 times, by each iteration
      3.1.1. Take randomly X% from the control database according to the
             percentage that corresponds to the iteration
      3.1.2. With that subgroup of cases of the control database, make the
             structural and parametric learning of RB-Complete and the RB- C4.5
      3.1.3. Evaluate the predictive power of both networks using the validation
             database
   3.2. Calculate the average predictive power (from the 30 iterations)
4. Graph the predictive power of both networks (RB-Complete and RB-C4.5) based on
   the cases of training
```

It is managed to minimize this effect, taking different data samples and average the obtained values. In the steps (3.1.x) it is made the structural and parametric learning of the RB-Complete and the RB-C4.5 from the subgroup of the control database (both networks are obtained from the same subgroup of data). Once obtained the network, it is come to evaluate the predictive capacity with the validation databases. This database is scan and for each row, all the evidence variables are instantiated and it is analyzed if the inferred class by the network corresponds with the indicated one in the file. Since the bayesian network does not make excluding classifications (it means that it predicts for each value of the class the probability of occurrence), is considered like the inferred class, the class with the greater probability. The predictive capacity corresponds to the percentage of cases classified correctly respect to the total evaluated cases. In the point (3.2) it is calculated the predictive power of the network, dividing the obtained values through all the made iterations. Finally, in the step (4) it is come to graph the predictive power average of both bayesian networks based on the amount of training cases.

3 Results

As it can be observed in Figure 1 (“Cancer” domain), the predictive power of the RB-C4.5 is superior to the one of RB-Complete throughout all its points. Also, it is possible to observe how this predictive capacity is increased, almost always, when it takes more cases of training to generate the networks. Finally, it is observed that from the 350 cases of training the predictive power of the networks become stabilized reaching its maximum level. When analyzing the graph of Figure 2 corresponding to the database “Cardiology”, also an improvement on the RB-C4.5 can be observed respect to RB-Complete. Although the differences between the values obtained with both networks are smaller that in the previous case, the hybrid algorithm presents a better approach to the reality that the other one.

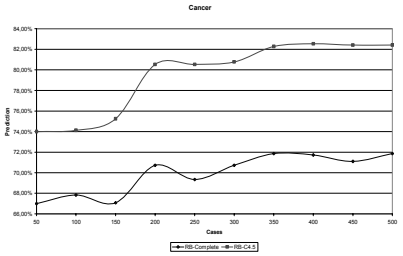


Fig. 1. Results on database "Cancer"

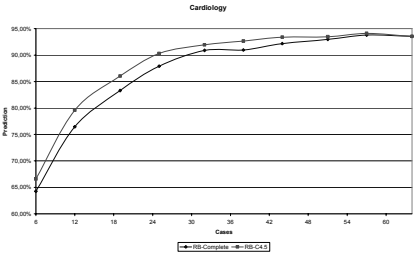


Fig. 2. Results on database "Cardiology"

4 Discussion and Conclusions

As it is possible to observe, all the graphs that represent the predictive power based on the amount of cases of training are increasing. This phenomenon occurs independently of the domain of data used and the evaluated method (RB-Complete or RB-C4.5). Of the analysis of the results obtained in the experimentation, we can (experimentally) conclude that the learning hybrid method used (RB-C4.5) generates an improvement in the predictive power of the network with respect to the obtained one without making the preprocessing of the variables (RB-Complete). In another aspect, the RB-C4.5 has a lesser amount of variables (or at the most equal) that RB-Complete, this reduction of the amount of involved variables produces a simplification of the analyzed domain, which carry out two important advantages; first, they facilitate the representation and interpretation of the knowledge removing parameters that do not concern on a direct way to the objective (classification task). Second, it simplifies and optimizes the reasoning task (propagation of the probabilities) which originates the improvement of the processing speed. In conclusion, from the obtained experimental results, we concluded that the hybrid learning method proposed in this paper optimizes the configurations of the bayesian networks in classification tasks.

References

1. Ramoni, M., Sebastiani, P. Bayesian *methods in Intelligent Data Analysis. An Introducction*. Pages 129-166. Physica Verlag, Heidelberg. (1999).
2. Diaz, F., Corchado, J. *Rough sets bases learning for bayesian networks*. International workshop on objetive bayesian methodology, Valencia, Spain. (1999).
3. Heckerman, D., Chickering, M. *Efficient approximation for the marginal likelihood of incomplete data given a bayesian network*. Technical report MSR-TR-96-08, Microsoft Research, Microsoft Corporation. (1996).
4. Murphy, P., Aha, D. *UCI Repository of Machine Learning databases. Machine-readable data repository*, http://mlearn.ics.uci.edu/MLRepository.html. Accessed March, 28th, (2007).

Application of Business Intelligence for Business Process Management

Nenad Stefanovic[1], Dusan Stefanovic[2], and Milan Misic[3]

Abstract Companies require highly automated business process management (BPM) functionality, with the flexibility to incorporate business intelligence (BI) at appropriate stages throughout the workflow. Business Activity Monitoring (BAM) unifies these two technologies and provides real-time access to critical performance indicators to improve the speed and effectiveness of business operations. This paper discusses BPM technologies in the context of the supply chain and presents the comprehensive BAM solution that utilizes latest BPM, BI and portal technologies in order to enable decision makers to access and assimilate the right information to make well-informed, timely decisions.

1 Introduction

Hammer [1] defines a business process as a complete set of end-to-end activities that together create value for the customer. Business Process Management (BPM) is the ability to orchestrate and control the execution of such processes across heterogeneous systems [2].

Business Process Management is the next step in a three-tier environment. Business logic and business rules, now encapsulated in the business logic tier, is extracted from the business logic tier and is presented in a workflow-based environment, which shows graphically the different steps of a business process. At each node, business rules are used to select the next node and business logic is executed. As a consequence, the business rules have become explicit, visible, and

[1] Nenad Stefanovic, MSc

Zastava Automobiles, Trg Topolivaca 4, Kragujevac, Serbia email: stefanovic.n@gmail.com

[2] Prof. Dusan Stefanovic

University of Kragujevac, Radoja Domanovica 12, Serbia email: dusans@kg.ac.yu

[3] Prof. Milan Misic

Higher Technical School, Zvecan, Serbia email: miskekm@eunet.rs

Please use the following format when citing this chapter:

Stefanovic, N., Stefanovic, D. and Misic, M., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 445449.

rapidly changeable. This allows a company to react more quickly on changes in the marketplace where it operates.

When BPMS (Business Process Management System) is used to solve business problems, the processing within those solutions is often a black box, making it very difficult for business users and technical support personnel to get a view into what is happening.

On the other hand, Business Intelligence (BI) tools rely on warehouses and data marts while event notification packages track business indicators, yet there's been little focus on merging data in real-time and from warehouses to track the enterprise's lifelines, maximize efficiency, and provide decision support data in context. This is where Business Activity Monitoring comes in. BAM enables any messaging or business process to be fully instrumented, monitored and analyzed in terms that any end-user can understand.

2 Business Activity Monitoring

Most, companies have no active real-time element to their BI systems. The consequences are that nothing is helping the business to automatically respond immediately when problems occur or opportunities arise. Also, there is no automatic notification or flagging of alerts to take action that may avoid unnecessary costs, business disruption, operational mistakes and unhappy customers in the future.

Business Activity Monitoring (BAM) is a collection of tools that allow you to manage aggregations, alerts, and profiles to monitor relevant business metrics (Key Performance Indicators - KPIs). It gives users end-to-end visibility into business processes, providing accurate information about the status and results of various operations, processes, and transactions so they can address problem areas and resolve issues within your business. BAM software products incorporate concepts from — and sometimes are built on — ERP, business intelligence, BPM and enterprise application integration (EAI) software.

The BAM provides an easy, real-time, transaction-consistent way to monitor heterogeneous business applications, and to present data for SQL queries and aggregated reports (OLAP). Through queries and aggregations BAM systems can include not only the data that is present during the running business process, but also the state and the dynamics of the running business process, independent of how the business is automated.

BAM applies operational business intelligence and application integration technologies to automated processes to continually refine them based on feedback that comes directly from knowledge of operational events [3]. In addition to auditing business processes (and business process management systems), BAM can send event-driven alerts that can be used to alert decision makers to changes in the business that may require action.

3 Example of the BPM solution

The need for automation and interaction of business processes necessitate the use of modern technologies for managing business process, trading partner relationships and monitoring and analyzing in real-time. For these purposes we have designed two specialized web portals - Business Activity Services (BAS) and Business Activities Monitoring (BAM) portal. This section presents the basis of the comprehensive BPM solution implemented in the automotive company.

3.1 Business Activity Services

Business Activity Services (BAS) provides an interaction and collaboration self-service portal Web site among supply network trading partners.

BAS provides the infrastructure to capture business user input into a business process easily. Then based on the human input, the business process (defined and automated as BPMS orchestration) can continue with the subsequent steps in the pre-defined workflow.

BAS web portal architecture consists of the following modules:

Business User Portal. The self-service Web site that enables business users to interact with partners and business processes through familiar metaphors such as Mailboxes.

Trading Partner Management (TPM). A set of interactive tools and forms that enable the business user to manage online interactions with trading partners. Figure 1 shows web page for Partner Profiles.

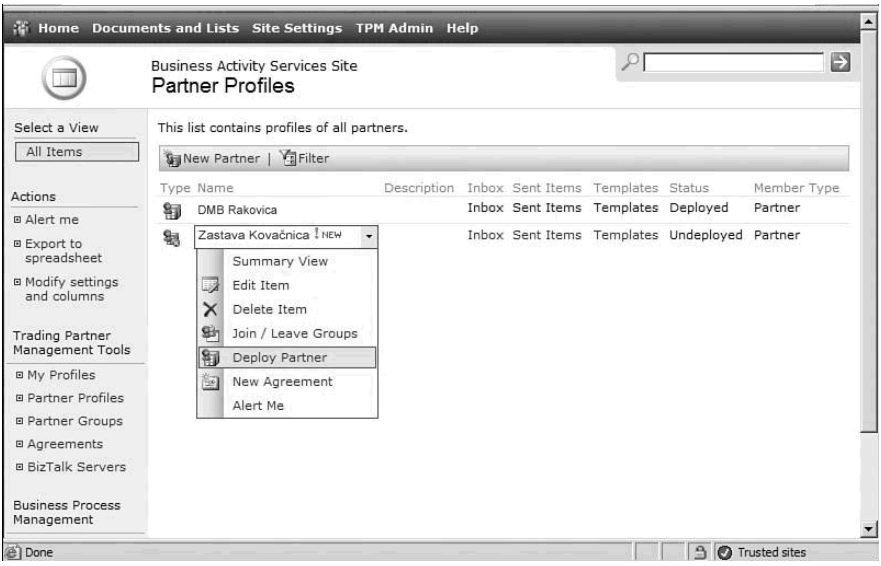


Figure 1 BAS Portal

Business Process Configuration. This primarily includes the design and programming of the Orchestrations and TPM elements on the BAS site in such a way that the business users can interact with them.

Business User Interaction and Collaboration (with partners and processes). As soon as the orchestrations and TPM elements go through the configuration process by using the key parameters, the business users can use the end-to-end infrastructure to perform the daily interactions with the trading partners.

3.2 BAM portal

Business users can use BAM portal to gain a real-time holistic view of business processes that span heterogeneous applications.

There are two ways information workers can use BAM to view business processes: using the spreadsheet application and through BAM web portal.

Each view gives a different perspective on a business process. For example, a BAM view might provide graphical depictions of per-product sales trends or current inventory levels or other key performance indicators. The information in these views might be updated every day, every hour, or more frequently.

Each BAM view relies on one or more BAM activities. A BAM activity represents a specific business process, such as handling purchase orders or shipping a product, and each one has a defined set of milestones and business data. For example, a purchase order activity might have milestones such as Approved, Denied, and Delivered along with business data like Customer Name and Product.

The following list describes how other ways information workers can use BAM features [4]:

- View a single activity instance such as a purchase order or loan (process) in real-time or as historical data.
- Search for activity instances based on their progress or business data (Figure 2).
- Browse aggregations (which are key performance indicators) around all the business activities that are currently being processed or have already happened.
- Navigate to the related activity instances such as shipments associated with given purchase order, or the Invoice in which it is included.

Additionally, it is possible to create different activity-related alerts. Alerts allow us to define important events about business processes, such as Key Performance Indicators (KPIs) that can be delivered to users on a real-time basis. Users subscribe to alerts to receive notification of the business event that the alert monitors. There are two types of alerts, aggregate and instance. An aggregate alert allows specifying threshold data across a time frame whereas an instance alert is based on specific qualifying data points.

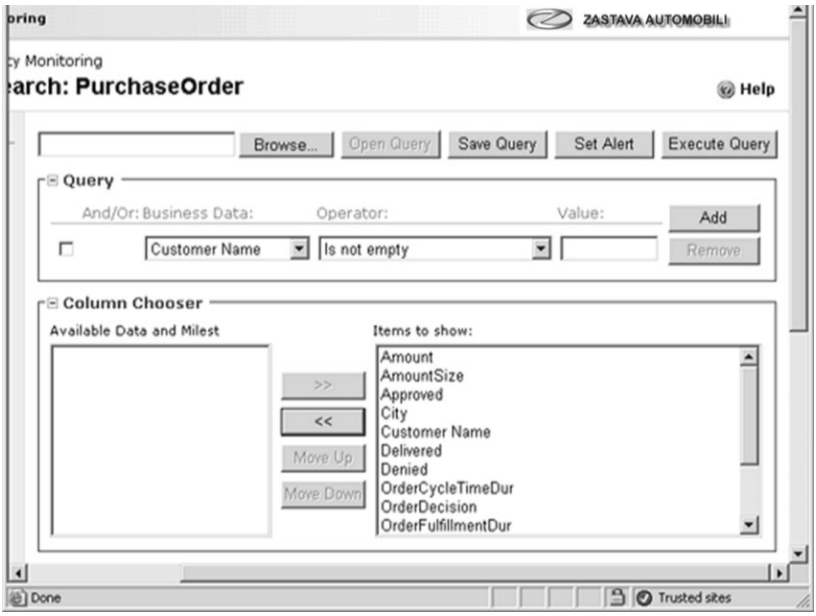


Figure 2 BAM search interface

4 Conclusion

This paper introduced the concept of Business Activity Monitoring and highlighted how it can be used to collect real-time information about business processes implemented through the BPMS.

The presented BAM portal provides a rich view into data collected via BAS system and enables activity data to be searched and viewed in a variety of ways. Because the BAM data is held in database tables and views, it's easy to access the information from a variety of tools, including different reporting tools, which can produce a highly detailed tracking portal providing very rich business intelligence.

With BPM and BAM systems in place, all parties in a supply chain network can track the real-time flow of goods, money, and information across the network.

References

[1] Hammer M, (1996) Beyond Reengineering: how the process-centered organization is changing our work and our lives. London: Harper Collins Business.
[2] Cutlip R, Telford R (2002) The Orchestration of Business Processes, Web Services Journal. 2(6): 28-34.
[3] Microsoft, (2006) What is BAM? http://msdn2.microsoft.com/en-us/library/aa560139.aspx.
[4] Stefanovic N, Stefanovic D (2006) Methodology for BPM in Supply Networks. 5th CIRP ICME, Ischia, Italy.

Learning Life Cycle in Autonomous Intelligent Systems

Jorge Ierache[1], Ramón García-Martínez[2], and Armando De Giusti[3]

Abstract Autonomous Intelligent Systems (AIS) integrate planning, learning, and execution in a closed loop, showing an autonomous intelligent behavior. A Learning Life Cycle (LLC) for AISs is proposed. The LLC is based on three different learned operators' layers: Built-In Operators, Trained Base Operators and World Interaction Operators. The extension of the original architecture to support the new type of operators is presented.

1 Introduction

The autonomous intelligent systems (AIS) evolve from initial theories (set of operators built in by the AIS's programmer) to ones learned from interaction with the environment or other. Given unknown environments, real autonomous systems must generate theories of how their environment reacts to their actions, and how the actions affect the environment. Usually, these learned theories are partial, incomplete and incorrect, but they can be used to plan, to further modify those theories, or to create new ones. Previous work on machine learning applied to problem solving has mainly focused on learning knowledge whose goal was to improve the efficiency of the problem solving task [1].
There is also a current interest in learning state transition probabilities in the context of reinforcement learning. However, few researchers have approached the generalized operators acquisition problem, described as techniques for automatically acquiring generalized descriptions of a domain theory. This issue is

[1] Jorge Ierache
PhD Program. Computer Sc. School. UNLP. Intelligent Systems Lab. FI-UBA. jierache@fi.uba.ar
[2] Ramon Garcia-Martinez
Software and Knowledge Engineering Center (CAPIS), Buenos Aires Institute of Technology
[3] Armando De Giusti
Instituto de Investigación en Informática LIDI, Facultad de Informática, UNLP

Please use the following format when citing this chapter:

Ierache, J., García-Martínez, R. and De Giusti, A., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and practice II*; Max Bramer; (Boston: Springer), pp. 451455.

crucial when dealing with systems that must autonomously adapt to an unknown and dynamic environment. LOPE (Learning by Observation in Planning Environments) is an AIS implemented architecture that integrates planning, learning, and execution in a closed loop, showing an autonomous intelligent behavior [2].

Learning planning operators (what we will call operators, is also referred to as action models within the reinforcement learning community) is achieved by observing the consequences of executing planned actions in the environment. In order to speed up the convergence, heuristic generalizations of the observations have been used. Also, probability distribution estimators have been introduced to handle the contradictions among the generated planning operators [3] and how sharing among AISs the learned operators improve their behavior [4].

As the natural next step, in this paper we recover the AIS general description (section 2), we propose the AIS learning life cycle for a community of AISs that shares knowledge (section 3) and preliminary conclusions and future research are drawn (section 4).

2 General description of AIS

One of the main objectives of each LOPE agent is to autonomously learn operators (action models) that predict the effects of actions in the environment by observing the consequences of those actions. In order to learn those descriptions, it is able to plan for achieving self-proposed goals, execute the plans, find out incorrect or correct behavior, and learn from the interaction with the environment and other agents. Each agent receives perceptions from the environment, called situations, applies actions, and learns from its interaction with the outside world (environment and other agents). At the beginning, the agent perceives the initial situation, and selects a random action to execute in the environment. Then, it loops by executing an action, perceiving the resulting situation and utility of the situation, learning from observing the effect of applying the action in the environment, and planning for further interactions with the environment when the previous plan has finished its execution, or the system observes a mismatch between the predicted situation by the AIS's operators and the situation it perceived from the environment. The planner basically does a backward chaining search from the initial situation (goal) of the operator with the highest utility in order to find a sequence of operators that will lead from the current state to that goal. If it succeeds, and the probability of its success is greater than a given bound, it executes the plan. If not, it selects the next highest utility operator and searches for a plan. This process loops until it finds a plan for any high utility operator. More details on how the planner works can be found in [3].

In this context, a learned operator O in LOPE [3] is a tuple <C,A,F,P,K,U> where: C is the initial situation (conditions), A action to be performed, F final situation (post-conditions), P times that the operator O_i was successfully applied

(the expected final situation F was obtained), K times that the action A was applied to C, U utility level reached applying the action to the initial situation C of the operator.

3 Proposed AISs Learning Life Cycle

Based on the LOPE Architecture, an AIS Learning Life Cycle with three learning layers is presented: [a] Layer BI (Built-In Operators) is the learning layer where the operators are implanted in the LOPE AIS by the AIS programmer, [b] layer TB (Trained Base Operators) is the learning layer where the operators are learned by training (previously designed by AIS programmer and evolutionary learning techniques) and [c] layer WI (World Interaction Operators) is the learning layer where the operators are learned by interaction with the part of the world that performs the environment of the AIS. The proposed learning life cycle is shown in figure 1.

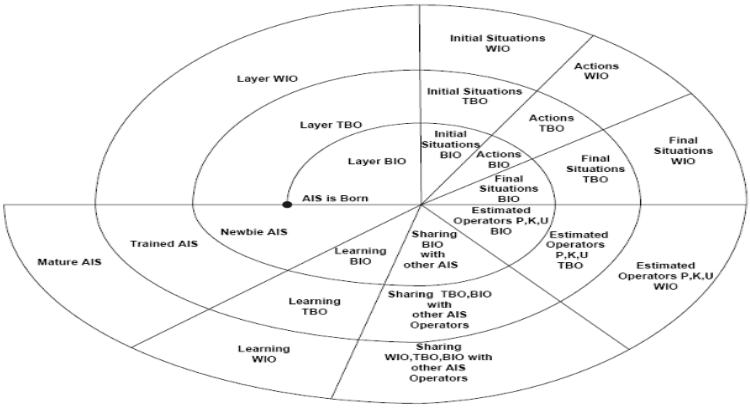


Fig. 1. AIS Learning Life Cycle (LLC)

Although knowledge sources are different, the sensor system and the learned operator structure is always the same. The AIS "born" with the implanted built in operators by its programmer. These operators represent the basal knowledge that allows an initial reactive behavior of the AIS. The operators learned by training facilitate the evolution of the knowledge using a reinforcement mechanism of good operators and the "punish" of bad functioning operators. The terms punish and reward have been borrowed from the field of biological reinforcement rather than from reinforcement learning. The heuristic-generalization algorithm generates a set of new operators according to the generalization heuristics, which are incorporated into the set of planning operators. Since the number of operators that are created can potentially slow down the performance of the learning and planning modules, the system forgets operators with a very low quotient P/K. One

of the main objectives of each LOPE-LLC (LOPE Learning Life Cycle) AIS is to autonomously learn operators (action models) that predict the effects of actions in the environment by observing the consequences of those actions. In order to learn those descriptions, it is able to plan for achieving self-proposed goals, executes the plans, find out incorrect or correct behavior, and learn from initial BI operators, reinforced by TB operators creation and evolved by knowledge sharing based WI operators from the interaction with the environment and other AISs. In TB operators learning each AIS receives perceptions from the environment, called situations, applies actions, and learns from its interaction in the designed training. The AIS perceives the initial situation based in its BI operators, and selects a random action to execute in the environment from its TB operators set. In the WI operators learning process use the BI operators and TB operators. Based on the three layers (BIO, BTO, WIO) of the LLC proposed, the AIS evolves by going around the stages, born, newbie, trained, mature. Each layer includes the following activities: [a] Initial situation of the world (environment and AIS's), [b] Actions based on the AIS operators according to their plans, [c] Foreseen final Situation, [d] Estimate of the AIS operators, [e] Operators sharing with other AISs and [f] AIS learning (regularly), [g] Evolution of the AIS into a new stage. When an AIS is born (initial stage), is provided with programmer's built-in operators. The BIO layer moves around this base, sharing its BI operators with other AIS's, learns and reaches the Newbie AIS stage and then it goes around the BTO layer learning through the training and sharing of the TB and BI operators which allow them to reach the Trained AIS stage, finally it goes around the WIO layer, capable to share their BI, BT, WI operators with the rest of the AIS's reaching the mature AIS stage. Figure 2 shows a schematic view of the LOPE architecture extended on the base of the LLC proposed, which is called LOPE-LLC.

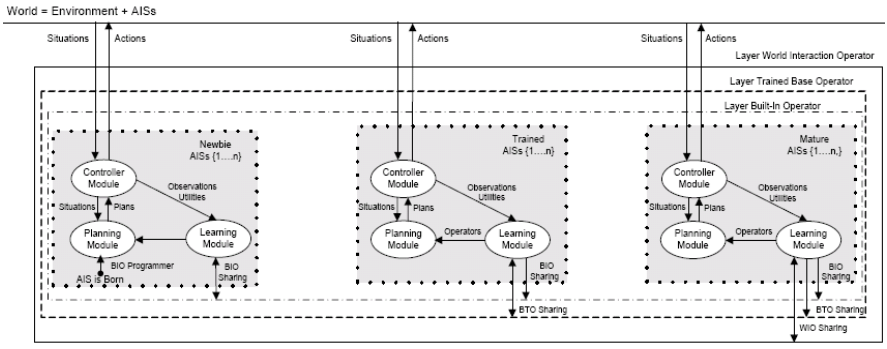


Fig. 2. Architecture of a group of LOPE –LLC AISs

LOPE-LLC AISs can be seen sharing its operators through the different layers according to their stage (born, newbie, trained, mature). Each of the AISs receives as input: situations (perceptions) from the world (environment and AISs); set of actions it can perform and Operators. The output of each AIS is a sequence of over time actions (for the environment) and regularly, the set of operators it learned

from the sharing with other AISs, according to the stage it reaches within the LLC layers proposed.

4 Preliminary Conclusions and Future Research

In this paper, we have presented a learning life cycle for autonomous intelligent systems based on three types of learned operators: built in operators, trained based operators and world interaction; and how these operators evolves in an architecture that learns a model of its environment by observing the effects of performing actions on it. The LOPE-LLC AISs autonomously interact with their environment and with other AISs with the objective of learning operators based on a proposed learning life cycle that predict, with a given probability estimator, the resulting situation of applying an action to another situation.

With respect to the scalability of the approach, we are now performing experiments in a much more complex, noisy, with hidden states, and many AIS domain, such as the robosoccer. The performance world is composed by the Environment (soccer field, ball) and the players of both teams (AIS's), programmed with operators to play the different roles (forward line players, midfield players, defenders, goalkeeper). BI operators of the player (AIS), resulting from the birth of this one thanks to the programmer's action, evolve while sharing with other players (AIS's) of their team or other roboccer teams, allowing the player (AIS) to reach the Newbie stage. BT operators (previously designed by AIS programmer and evolutionary learning techniques) make easier the (AIS) player's evolution into the trained stage, after finishing the activities of the BTO layer of the LLC. The trained player (AIS) through the WI operators sharing with other players (AIS's of its team or another team) reaches the mature player (AIS) stage once the activities of the WIO layer of LLC are finished. We believe that through the use of the probabilities estimations, and the heuristic generalization of operators, we will be able to cope with the complexity of that domain.

References

1. Fritz, W., García-Martínez, R., Blanqué, J., Rama, A., Adobbati, R., Sarno, M. *The autonomous intelligent system*. Robotics and Autonomous Systems, 5, 109-125. (1989).
2. García-Martínez, R. and Borrajo, D. *Planning, learning, and executing in autonomous systems*. Lecture Notes in Artificial Intelligence, 1348, 208-220. (1997).
3. García Martínez, R. y Borrajo, D. *An Integrated Approach of Learning, Planning and Executing*. Journal of Intelligent and Robotic Systems, 29, 47-78. (2000).
4. García-Martínez, R., Borrajo, D., Britos, P., Maceri, P. *Learning by Knowledge Sharing in Autonomous Intelligent Systems*. Lecture Notes in Artificial Intelligence, 4140, 128-137. (2006).

A Map-based Integration of Ontologies into an Object-Oriented Programming Language

Kimio Kuramitsu

Abstract Today's programmers have difficulties using ontology in their information-centric applications, where ontology would be useful. This paper addresses the integration technique of ontologies into an object-oriented scripting language. Our technique is based on the use of semantic mapping as a unified form of complicated semantic relations in an ontology system for the class-subclass view of an object-oriented programming modeling. This enables ordinary programmers to write ontology reasoning, such as equivalence and subsumption, without any extended logical constructors.

1 Introduction

The ontology technology has been widely accepted as an integral part of managing the semantics of information on the Web and other information centric systems [3]. More recently, with the popularity of the Semantic Web, practical ontology languages and tools, such as Jena [4], have been developed to share ontology through the Web. Despite of these growing concerns, there is still a huge difficulty receiving ontology benefits, especially for most of programmers who are developing web and information-rich applications where ontology would be potentially helpful.

One considerable reason is that the terminology of ontology is quit different from that of object-oriented programming languages that today's developers are very familiar with. Developers who want to use some APIs in an ontology tool, such as Jena or Fact++, have to learn about logical constructors to use the existing ontology, because they are mainly designed for KR experts to build their ontologies.

The purpose of this paper is to present a map-based approach to integrating the use of ontology into well-known constructors in an object-oriented programming language. In our approach, concepts and individuals are transparently mapped to classes and its instances, and semantic reasoning such as *equivalence* and *subsump-*

Kimio Kuramitsu
Yokohama National University, Yokohama City, Japan, e-mail: kimio@ynu.ac.jp This work has been supported in part by Grant-in-Aid for Japanese Scientific Research (1870002300) and SCOPE-R funds (062103013).

Please use the following format when citing this chapter:

Kuramitsu, K., 2008, in IFIP International Federation for Information Processing, Volume 276; *Artificial Intelligence and Practice II*; Max Bramer; (Boston: Springer), pp. 457461.

tion can be operated with new operators === or `isa`, which would be as friendly as `instanceof`. This enables us to write semantic program naturally like:

```
Medicine m = "Amoxillin";
if(m isa Antibiotics || m === "Penicillin")  ..
```

The strength of our map-based approach is in its ontology-language neutrality. We use *semantic mapping* as a unified view to redefine complicated conceptual relations in an ontology system. This allows us to use any type of classification-based knowledge as a part of programmed codes without external logical operators.

We will show the map-based integration through our implemented scripting language, Konoha[1]. Section 2 is an introduction of the use of ontologies in Konoha. In Section 3, we define the semantic mapping that mediates two different worlds: the ontology and the object-oriented modeling. In Section 4, we will review related work. In Section 5, we conclude the paper.

2 Use of Ontologies with Konoha

Every programming language has primitive types, such as int and String, which are used to represent very basic information values. However, they cannot carry any semantics that identify the concepts of its information. For example, the class String is available to represent a name of person, email, ISBN, and even an arbitrary plain text, while it provides no help for identifying the meaning of its represented string. Konoha allows us to extend primitive types, such as Int, Float, String, by adding semantic identifiers, URN (Universal Resource Name). The using statement is newly introduced to add a class to URN-specified semantic constraints.

Here is the first example, where the meaning of Celsius is added into Float. A new class, named Float::C, is generated as a result, and its instance value is associted with its semantics through the URN. (Note that, Float::C is a local name and, in global, the class is identified with URN.)

```
>>> using Float::C  http://unit/Celsius
>>> Float::C t = 20;
>>> t
20[C]
>>> t.class
Float{http://unit/Celsius}
```

Next, we suppose a vocabulary set, which is used to represent feeling temperature such as “freezing”, “chilly”, “cool”, “comfortable”.

```
>>> using String::feel  http://vocabulary/FeelTemp
>>> String::feel ft = "feel:chilly";
>>> ft = "hello,world";       (==> InvaidValueException)
```

The class String::feel is not only semantically annotated, but also constrained in the range of its instance values. The String::feel allows to take vocabulary strings that are specified in `http://vocabulary/FeelTemp`.

[1] Our first prototyped implementation of Konoha is downloadable at http://konoha.sourceforge.jp/.

The semantic-extended class, although it is helpful for programmers to remember its meanings, is still meaningless in machine processing. That is, Konoha is able to know that Float::C and String::feel are different, but not to know whether 20C is "comfortable" or not. To obtain such a question, a reasoning system will be needed here.

In Konoha, reasoning is a part of casting/mapping between two classes. If the programmer wants to know whether 20C is comfortable, he or she can simply write as follows:

```
>>> t = 20C;
>>> (String::feel)t
"comfortable"
```

Konoha has no its own reasoning system. When it recieves a request through the mapping operation, it poses a map-based query, say, $? : -20C \mapsto \mathsf{String} :: \mathsf{feel}$ for an external ontology system, which the associated URNs indicate to. Due to the unified form of querying/answering, there is no additional library to connect the external system.

3 Bridging Two Worlds

3.1 Class and Concept

The class, in an OO world, and the concept, in the KR world, are very similar, but they differ in that a class is specified first and its objects are instantiated after the class definition while individuals exist at first and its concept is reasoned later by classification.

As our starting point, we have chosen to build the KR concept on top of the class-first world. That is, all individuals are belonging to one existing concept from the beginning. Let C be a concept name. We write C^I for a set of individuals that belongs to C. We say $\mathtt{t} \in C^I$ if a given $\mathtt{t}$ is an instance of C.

Here are examples of defining two concepts AmericanSeason and BritishSeason.

$$\mathsf{AmericanSeason}^I = \{\mathtt{spring}, \mathtt{summer}, \mathtt{fall}, \mathtt{winter}\}$$
$$\mathsf{BritishSeason}^I = \{\mathtt{spring}, \mathtt{summer}, \mathtt{autumn}, \mathtt{winter}\}$$

These two concepts seem to be very similar, because both of them have the same individuals, such as `spring`, `summer`, and `winter`. However, by default, we regard these individuals as *homonyms*, i.e., the same symbols having different meanings. To identify conceptual differences between individuals, we write an instance $C.\mathtt{t}$ for $\mathtt{t} \in C^I$.

3.2 Semantic Mapping

Between two concepts, there is no semantic relation by default. To add semantic relation, we use *semantic mapping*, denoted $C \mapsto D$.

To begin with, we focus on two instances $C.x$ and $D.y$. We say $C.x \mapsto D.y$ if $C.x$ is interpreted as $D.y$, the concept of $C.x$ is *broader* than that of $D.y$, or, from the perspective of relative information capacity [6], $C.x$ is more informative than $D.y$.

In addition, we say $C.x$ and $D.y$ is semantically equivalent, denoted $C.x \equiv D.y$, if and only if $C.x \mapsto D.y$ and $D.y \mapsto C.x$.

Next, we will extend the semantic mapping from two indeviduals to two concepts.

Definition 1 (semantic mapping and equivalence)

$$\frac{\forall x\, \exists y\, C.x \mapsto D.y}{C \mapsto D}, \quad \frac{C \mapsto D \qquad D \mapsto C}{C \equiv D} \tag{1}$$

Note that for simplicity all semantic mappings in this paper are supposed to be *total*, although *partial mappings* would be very common. In practice, we use null, the null pointer widely used in programming languages, to represent a partial mapping. We say no mapping if $C.x \mapsto \mathsf{null}$, and we write $C \not\mapsto D$ if for each $x \in C^I$ $C.x \mapsto D.\mathsf{null}$. The class C, D are disjoint if $C \not\mapsto D$ and $D \not\mapsto C$.

3.3 Subtyping System

The *subtyping* system, generally supported in object-oriented programming languages, allows us to organize classes in a class-subclass manner. We use a partial order to represent the organized class-subclass relation; we write $C \prec D$ for the class declaration.

Konoha has the same grammar and transitivity property with Java for subtyping.

$$\frac{\mathsf{class}\ C\ \mathsf{extends}\ D\ \{...\}}{C \prec D}, \quad \frac{C \prec D \qquad D \prec E}{C \prec E} \tag{2}$$

3.4 Bridging Ontology

An ontology is a set of *structured* terms. The "structure" is given by mathematical relations, like $\mathsf{C}(t)$ and $\mathsf{R}(t,t2)$. which are called respectively *concept* and *role*. Although different class of ontology languages [1] introduce different variation of roles, from the classification view they comonly provides three types of reasoned relations.

- (*equivalence*) $C \equiv D$,
- (*subsumption*) $C \sqsubseteq D$
- (*disjointness*) $C \sqcap D = \bot$

Note that we are interested only in these three relations due to the similarity with the class-subclass relation in object-oriented programming languages.

Theorem 1. *Our concept definition and semantic mapping contain $C \equiv D$, $C \sqsubseteq D$, and $C \sqcap D = \bot$.*

Proof(sketch). Let Δ be a finite set of terms in an ontology system. Suppose $t \in \Delta$. If a unary relation $C(t)$ is true, then we make a new instance $C.t$ in C^I. We always say $C.t \equiv D.t$ because t is identical on Δ. On the other hand, $C(t)$ is said to be true

if $C \sqsubseteq D$ and $D(t)$ is true. Accordingly, we say $C.t \mapsto D.t$ for all t that satisfies both $C(t)$ and $C \sqsubseteq D$ (, i.e., $D(t)$ is true).

4 Related Work

There is a long history of representing knowledge representation in a LISP-style syntax. It is not unnatural to combine deductive programming features, such as Prolog, with such a LISP-style ontology description, or vice versa. More recently, Go! [2] was designed to integrate an object-oriented prolog with its own ontology description. However, the integration of a logic-based programming language with ontology constructors requires different elaborations. ActiveRDF [7] showed an ORM-style approach to the integration of RDF with Ruby, where Ruby classes are generated dynamically by SPARQL queries. This enables us to use RDF/S semantics transparently in Ruby classes. However, their mapping method is so direct that it cannot map more reasoned relations, such as equivalence and subsumption.

5 Conclusion

Today's programmers have difficulties using ontology in their information-centric applications, where ontology would be useful. This paper addressed the map-based integration of ontologies into an object-oriented scripting language. Our technique is based on semantic mapping, a unified form of complicated semantic relations in an ontology system for class-subclass view of an object-oriented programming modeling. Using Konoha, we showed a programmer is able to write ontology reasoning, such as equivalence and subsumption, without any extended logical constructors.

References

1. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider (eds). *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2000.
2. Keith L. Clark and Frank G. McCabe Ontology Oriented Programming in Go! Journal of Applied Intelligence, 2005.
3. Michael Gruninger and Jintae Lee. Special issue: Ontology applications and design. *Communications of the ACM*, 45(2):39–41, 2002.
4. Jena - A Semantic Web Framework for Java. http://jena.sourceforge.net/
5. K. Kuramitsu. Mappings As A Lightweight Ontology System for the World-Wide Web. In *Proc. of the Symposium on Professional Practice in AI / IFIP World Computer Congress (WCC2004)*, 2004.
6. Renée J. Miller, Yannis E. Ioannidis, and Raghu Ramakrishnan. The use of information capacity in schema integration and translation. In *Proceedings of 19th International Conference on Very Large Data Bases*, pages 120–133. Morgan Kaufmann, 1993.
7. Eyal Oren, Renaud Delbru, Sebastian Gerke, Armin Haller, and Stefan Decker. ActiveRDF: Object-Oriented Semantic Web Programming. In *Proc. of WWW2007*, 2007.
8. Peter F. Patel-Schneider, Patrick Hayes, and Ian Horrocks(eds.) OWL Web Ontology Language: Semantics and Abstract Syntax, *W3C Recommendation*, 10 February, 2004.