consequence, any Web user can push his personal agenda with little effort and almost at no cost. This universe without frontiers has attracted tremendous attention from millions of people everywhere since the very beginning. Furthermore, it is causing a revolution in the way people use computers and perform their daily tasks. For instance, home shopping and home banking are becoming very popular and have generated several hundred million dollars in revenues.

Despite so much success, the Web has introduced new problems of its own. Finding useful information on the Web is frequently a tedious and difficult task. For instance, to satisfy his information need, the user might navigate the space of Web links (i.e., the *hyperspace*) searching for information of interest. However, since the hyperspace is vast and almost unknown, such a navigation task is usually inefficient. For naive users, the problem becomes harder, which might entirely frustrate all their efforts. The main obstacle is the absence of a well defined underlying data model for the Web, which implies that information definition and structure is frequently of low quality. These difficulties have attracted renewed interest in IR and its techniques as promising solutions. As a result, almost overnight, IR has gained a place with other technologies at the center of the stage.

1.1.3 Focus of the Book

Despite the great increase in interest in information retrieval, modern textbooks on IR with a broad (and extensive) coverage of the various topics in the field are still difficult to find. In an attempt to partially fulfill this gap, this book presents an overall view of research in IR from a computer scientist's perspective. This means that the focus of the book is on computer algorithms and techniques used in information retrieval systems. A rather distinct viewpoint is taken by librarians and information science researchers, who adopt a human-centered interpretation of the IR problem. In this interpretation, the focus is on trying to understand how people interpret and use information as opposed to how to structure, store, and retrieve information automatically. While most of this book is dedicated to the computer scientist's viewpoint of the IR problem, the human-centered viewpoint is discussed to some extent in the last two chapters.

We put great emphasis on the integration of the different areas which are closed related to the information retrieval problem and thus, should be treated together. For that reason, besides covering text retrieval, library systems, user interfaces, and the Web, this book also discusses visualization, multimedia retrieval, and digital libraries.

1.2 Basic Concepts

The effective retrieval of relevant information is directly affected both by the *user task* and by the *logical view of the documents* adopted by the retrieval system, as we now discuss.

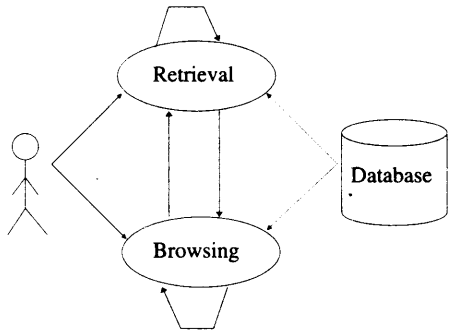

Figure 1.1 Interaction of the user with the retrieval system through distinct tasks.

1.2.1 The User Task

The user of a retrieval system has to translate his information need into a query in the language provided by the system. With an information retrieval system, this normally implies specifying a set of words which convey the semantics of the information need. With a data retrieval system, a query expression (such as, for instance, a regular expression) is used to convey the constraints that must be satisfied by objects in the answer set. In both cases, we say that the user searches for useful information executing a *retrieval* task.

Consider now a user who has an interest which is either poorly defined or which is inherently broad. For instance, the user might be interested in documents about car racing in general. In this situation, the user might use an interactive interface to simply look around in the collection for documents related to car racing. For instance, he might find interesting documents about Formula 1 racing, about car manufacturers, or about the '24 Hours of Le Mans.' Furthermore, while reading about the '24 Hours of Le Mans', he might turn his attention to a document which provides directions to Le Mans and, from there, to documents which cover tourism in France. In this situation, we say that the user is *browsing* the documents in the collection, not searching. It is still a process of retrieving information, but one whose main objectives are not clearly defined in the beginning and whose purpose might change during the interaction with the system.

In this book, we make a clear distinction between the different tasks the user of the retrieval system might be engaged in. His task might be of two distinct types: information or data *retrieval* and *browsing*. Classic information retrieval systems normally allow information or data retrieval. Hypertext systems are usually tuned for providing quick browsing. Modern digital library and Web interfaces might attempt to combine these tasks to provide improved retrieval capabilities. However, combination of retrieval and browsing is not yet a well

established approach and is not the dominant paradigm.

Figure 1.1 illustrates the interaction of the user through the different tasks we identify. Information and data retrieval are usually provided by most modern information retrieval systems (such as Web interfaces). Further, such systems might also provide some (still limited) form of browsing. While combining information and data retrieval with browsing is not yet a common practice, it might become so in the future.

Both retrieval and browsing are, in the language of the World Wide Web, 'pulling' actions. That is, the user requests the information in an interactive manner. An alternative is to do retrieval in an automatic and permanent fashion using software agents which *push* the information towards the user. For instance, information useful to a user could be extracted periodically from a news service. In this case, we say that the IR system is executing a particular retrieval task which consists of *filtering* relevant information for later inspection by the user. We briefly discuss filtering in Chapter 2.

1.2.2 Logical View of the Documents

Due to historical reasons, documents in a collection are frequently represented through a set of index terms or keywords. Such keywords might be extracted directly from the text of the document or might be specified by a human subject (as frequently done in the information sciences arena). No matter whether these representative keywords are derived automatically or generated by a specialist, they provide a *logical view of the document.* For a precise definition of the concept of a document and its characteristics, see Chapter 6.

Modern computers are making it possible to represent a document by its full set of words. In this case, we say that the retrieval system adopts a *full text* logical view (or representation) of the documents. With very large collections, however, even modern computers might have to reduce the set of representative keywords. This can be accomplished through the elimination of *stopwords* (such as articles and connectives), the use of *stemming* (which reduces distinct words to their common grammatical root), and the identification of noun groups (which eliminates adjectives, adverbs, and verbs). Further, compression might be employed. These operations are called *text operations* (or transformations) and are covered in detail in Chapter 7. Text operations reduce the complexity of the document representation and allow moving the logical view from that of a full text to that of a set of *index terms.*

The full text is clearly the most complete logical view of a document but its usage usually implies higher computational costs. A small set of categories (generated by a human specialist) provides the most concise logical view of a document but its usage might lead to retrieval of poor quality. Several intermediate logical views (of a document) might be adopted by an information retrieval system as illustrated in Figure 1.2. Besides adopting any of the intermediate representations, the retrieval system might also recognize the internal structure normally present in a document (e.g., chapters, sections, subsections, etc.). This

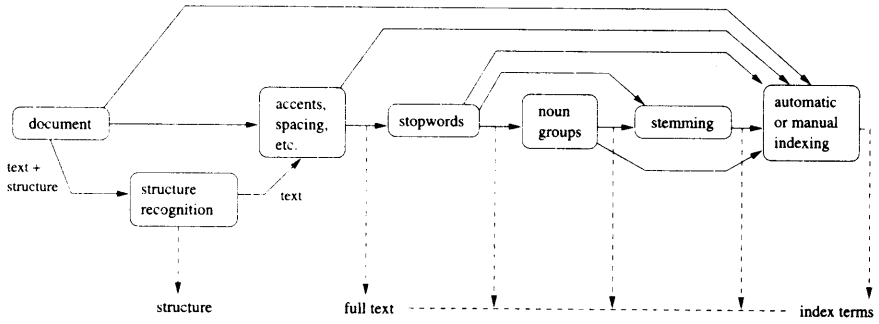

Figure 1.2 Logical view of a document: from full text to a set of index terms.

information on the structure of the document might be quite useful and is required by structured text retrieval models such as those discussed in Chapter 2.

As illustrated in Figure 1.2, we view the issue of logically representing a document as a continuum in which the logical view of a document might shift (smoothly) from a full text representation to a higher level representation specified by a human subject.

1.3 Past, Present, and Future

1.3.1 Early Developments

For approximately 4000 years, man has organized information for later retrieval and usage. A typical example is the table of contents of a book. Since the volume of information eventually grew beyond a few books, it became necessary to build specialized data structures to ensure faster access to the stored information. An old and popular data structure for faster information retrieval is a collection of selected words or concepts with which are associated pointers to the related information (or documents) — the *index*. In one form or another, indexes are at the core of every modern information retrieval system. They provide faster access to the data and allow the query processing task to be speeded up. A detailed coverage of indexes and their usage for searching can be found in Chapter 8.

For centuries, indexes were created manually as categorization hierarchies. In fact, most libraries still use some form of categorical hierarchy to classify their volumes (or documents), as discussed in Chapter 14. Such hierarchies have usually been conceived by human subjects from the library sciences field. More recently, the advent of modern computers has made possible the construction of large indexes automatically. Automatic indexes provide a view of the retrieval problem which is much more related to the system itself than to the user need.

In this respect, it is important to distinguish between two different views of the IR problem: a computer-centered one and a human-centered one.

In the computer-centered view, the IR problem consists mainly of building up efficient indexes, processing user queries with high performance, and developing ranking algorithms which improve the 'quality' of the answer set. In the human-centered view, the IR problem consists mainly of studying the behavior of the user, of understanding his main needs, and of determining how such understanding affects the organization and operation of the retrieval system. According to this view, keyword based query processing might be seen as a strategy which is unlikely to yield a good solution to the information retrieval problem in the long run.

In this book, we focus mainly on the computer-centered view of the IR problem because it continues to be dominant in the market place.

1.3.2 Information Retrieval in the Library

Libraries were among the first institutions to adopt IR systems for retrieving information. Usually, systems to be used in libraries were initially developed by academic institutions and later by commercial vendors. In the first generation, such systems consisted basically of an automation of previous technologies (such as card catalogs) and basically allowed searches based on author name and title. In the second generation, increased search functionality was added which allowed searching by subject headings, by keywords, and some more complex query facilities. In the third generation, which is currently being deployed, the focus is on improved graphical interfaces, electronic forms, hypertext features, and open system architectures.

Traditional library management system vendors include Endeavor Information Systems Inc., Innovative Interfaces Inc., and EOS International. Among systems developed with a research focus and used in academic libraries, we distinguish Okapi (at City University, London), MELVYL (at University of California), and Cheshire II (at UC Berkeley). Further details on these library systems can be found in Chapter 14.

1.3.3 The Web and Digital Libraries

If we consider the search engines on the Web today, we conclude that they continue to use indexes which are very similar to those used by librarians a century ago. What has changed then?

Three dramatic and fundamental changes have occurred due to the advances in modern computer technology and the boom of the Web. First, it became a lot cheaper to have access to various sources of information. This allows reaching a wider audience than ever possible before. Second, the advances in all kinds of digital communication provided greater access to networks. This implies that the information source is available even if distantly located and that

the access can be done quickly (frequently, in a few seconds). Third, the freedom to post whatever information someone judges useful has greatly contributed to the popularity of the Web. For the first time in history, many people have free access to a large publishing medium.

Fundamentally, low cost, greater access, and publishing freedom have allowed people to use the Web (and modern digital libraries) as a highly *interactive* medium. Such interactivity allows people to exchange messages, photos, documents, software, videos, and to 'chat' in a convenient and low cost fashion. Further, people can do it at the time of their preference (for instance, you can buy a book late at night) which further improves the convenience of the service. Thus, high interactivity is the fundamental and current shift in the communication paradigm. Searching the Web is covered in Chapter 13, while digital libraries are covered in Chapter 15.

In the future, three main questions need to be addressed. First, despite the high interactivity, people still find it difficult (if not impossible) to retrieve information relevant to their information needs. Thus, in the dynamic world of the Web and of large digital libraries, which techniques will allow retrieval of higher quality? Second, with the ever increasing demand for access, quick response is becoming more and more a pressing factor. Thus, which techniques will yield faster indexes and smaller query response times? Third, the quality of the retrieval task is greatly affected by the user interaction with the system. Thus, how will a better understanding of the user behavior affect the design and deployment of new information retrieval strategies?

1.3.4 Practical Issues

Electronic commerce is a major trend on the Web nowadays and one which has benefited millions of people. In an electronic transaction, the buyer usually has to submit to the vendor some form of credit information which can be used for charging for the product or service. In its most common form, such information consists of a credit card number. However, since transmitting credit card numbers over the Internet is not a safe procedure, such data is usually transmitted over a fax line. This implies that, at least in the beginning, the transaction between a new user and a vendor requires executing an off-line procedure of several steps before the actual transaction can take place. This situation can be improved if the data is encrypted for security. In fact, some institutions and companies already provide some form of encryption or automatic authentication for security reasons.

However, security is not the only concern. Another issue of major interest is privacy. Frequently, people are willing to exchange information as long as it does not become public. The reasons are many but the most common one is to protect oneself against misuse of private information by third parties. Thus, privacy is another issue which affects the deployment of the Web and which has not been properly addressed yet.

Two other very important issues are copyright and patent rights. It is far

from clear how the wide spread of data on the Web affects copyright and patent laws in the various countries. This is important because it affects the business of building up and deploying large digital libraries. For instance, is a site which supervises all the information it posts acting as a publisher? And if so, is it responsible for a misuse of the information it posts (even if it is not the source)?

Additionally, other practical issues of interest include scanning, optical character recognition (OCR), and cross-language retrieval (in which the query is in one language but the documents retrieved are in another language). In this book, however, we do not cover practical issues in detail because it is not our main focus. The reader interested in details of practical issues is referred to the interesting book by Lesk [501].

1.4 The Retrieval Process

At this point, we are ready to detail our view of the retrieval process. Such a process is interpreted in terms of component subprocesses whose study yields many of the chapters in this book.

To describe the retrieval process, we use a simple and generic software architecture as shown in Figure 1.3. First of all, before the retrieval process can even be initiated, it is necessary to define the text database. This is usually done by the manager of the database, which specifies the following: (a) the documents to be used, (b) the operations to be performed on the text, and (c) the text model (i.e., the text structure and what elements can be retrieved). The text operations transform the original documents and generate a logical view of them.

Once the logical view of the documents is defined, the database manager (using the DB Manager Module) builds an index of the text. An index is a critical data structure because it allows fast searching over large volumes of data. Different index structures might be used, but the most popular one is the *inverted file* as indicated in Figure 1.3. The resources (time and storage space) spent on defining the text database and building the index are amortized by querying the retrieval system many times.

Given that the document database is indexed, the retrieval process can be initiated. The user first specifies a *user need* which is then parsed and transformed by the same text operations applied to the text. Then, *query operations* might be applied before the actual *query*, which provides a system representation for the user need, is generated. The query is then processed to obtain the *retrieved documents*. Fast query processing is made possible by the index structure previously built.

Before been sent to the user, the retrieved documents are ranked according to a *likelihood* of relevance. The user then examines the set of ranked documents in the search for useful information. At this point, he might pinpoint a subset of the documents seen as definitely of interest and initiate a *user feedback* cycle. In such a cycle, the system uses the documents selected by the user to change the query formulation. Hopefully, this modified query is a better representation

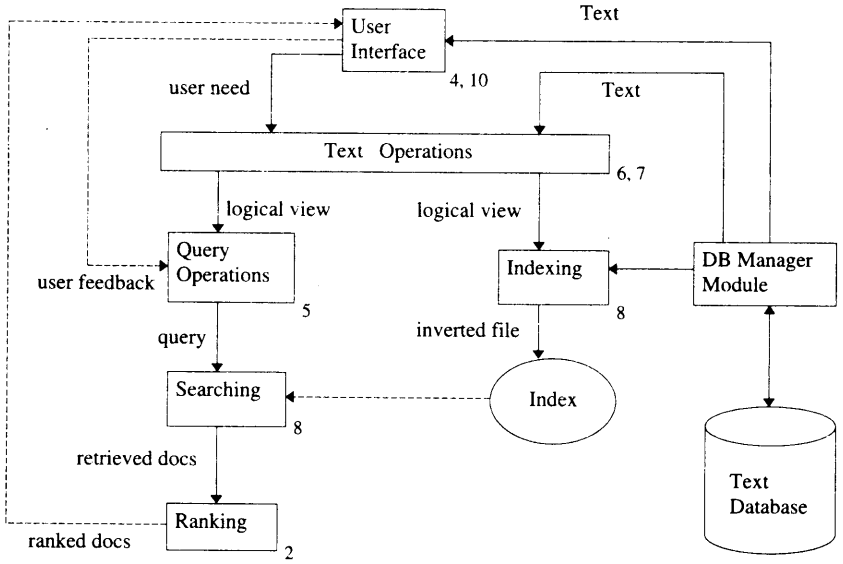

Figure 1.3 The process of retrieving information (the numbers beside each box indicate the chapters that cover the corresponding topic).

of the real user need.

The small numbers outside the lower right corner of various boxes in Figure 1.3 indicate the chapters in this book which discuss the respective subprocesses in detail. A brief introduction to each of these chapters can be found in section 1.5.

Consider now the user interfaces available with current information retrieval systems (including Web search engines and Web browsers). We first notice that the user almost never declares his information need. Instead, he is required to provide a direct representation for the query that the system will execute. Since most users have no knowledge of text and query operations, the query they provide is frequently inadequate. Therefore, it is not surprising to observe that poorly formulated queries lead to poor retrieval (as happens so often on the Web).

1.5 Organization of the Book

For ease of comprehension, this book has a straightforward structure in which four main parts are distinguished: text IR, human-computer interaction (HCI)

for IR, multimedia IR, and applications of IR. *Text IR* discusses the classic problem of searching a collection of documents for useful information. *HCI for IR* discusses current trends in IR towards improved user interfaces and better data visualization tools. *Multimedia IR* discusses how to index document images and other binary data by extracting features from their content and how to search them efficiently. On the other hand, document images that are predominantly text (rather than pictures) are called *textual images* and are amenable to automatic extraction of keywords through metadescriptors, and can be retrieved using text IR techniques. *Applications of IR* covers modern applications of IR such as the Web, bibliographic systems, and digital libraries. Each part is divided into topics which we now discuss.

1.5.1 Book Topics

The four parts which compose this book are subdivided into eight topics as illustrated in Figure 1.4. These eight topics are as follows.

The topic *Retrieval Models & Evaluation* discusses the traditional models of searching text for useful information and the procedures for evaluating an information retrieval system. The topic *Improvements on Retrieval* discusses techniques for transforming the query and the text of the documents with the aim of improving retrieval. The topic *Efficient Processing* discusses indexing and searching approaches for speeding up the retrieval. These three topics compose the first part on Text IR.

The topic *Interfaces & Visualization* covers the interaction of the user with the information retrieval system. The focus is on interfaces which facilitate the process of specifying a query and provide a good visualization of the results.

The topic *Multimedia Modeling & Searching* discusses the utilization of multimedia data with information retrieval systems. The focus is on modeling, indexing, and searching multimedia data such as voice, images, and other binary data.

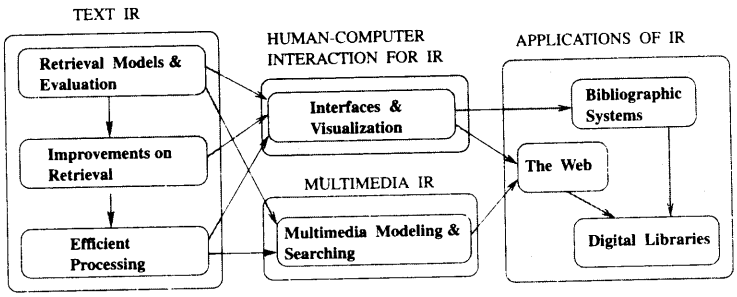

Figure 1.4 Topics which compose the book and their relationships.

The part on applications of IR is composed of three interrelated topics: *The Web*, *Bibliographic Systems*, and *Digital Libraries*. Techniques developed for the first two applications support the deployment of the latter.

The eight topics distinguished above generate the 14 chapters, besides this introduction, which compose this book and which we now briefly introduce.

1.5.2 Book Chapters

Figure 1.5 illustrates the overall structure of this book. The reasoning which yielded the chapters from 2 to 15 is as follows.

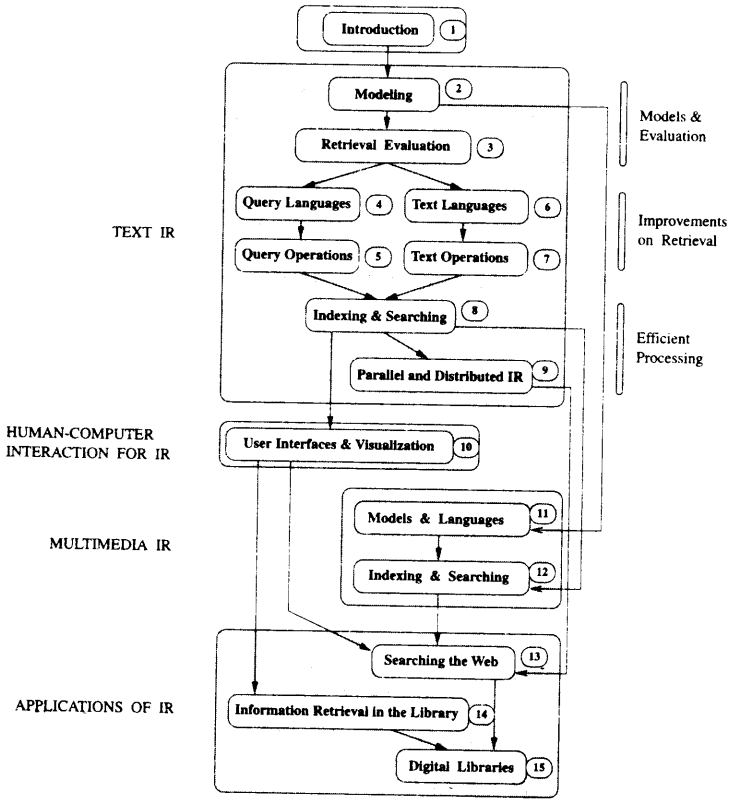

Figure 1.5 Structure of the book.

In the traditional keyword-based approach, the user specifies his information need by providing sets of keywords and the information system retrieves the documents which best approximate the user query. Also, the information system

might attempt to rank the retrieved documents using some measure of relevance. This ranking task is critical in the process of attempting to satisfy the user information need and is the main goal of *modeling* in IR. Thus, information retrieval models are discussed early in Chapter 2. The discussion introduces many of the fundamental concepts in information retrieval and lays down much of the foundation for the subsequent chapters. Our coverage is detailed and broad. Classic models (Boolean, vector, and probabilistic), modern probabilistic variants (belief network models), alternative paradigms (extended Boolean, generalized vector, latent semantic indexing, neural networks, and fuzzy retrieval), structured text retrieval, and models for browsing (hypertext) are all carefully introduced and explained.

Once a new retrieval algorithm (maybe based on a new retrieval model) is conceived, it is necessary to evaluate its performance. Traditional evaluation strategies usually attempt to estimate the costs of the new algorithm in terms of time and space. With an information retrieval system, however, there is the additional issue of evaluating the relevance of the documents retrieved. For this purpose, text reference collections and evaluation procedures based on variables other than time and space are used. Chapter 3 is dedicated to the discussion of *retrieval evaluation.*

In traditional IR, queries are normally expressed as a set of keywords which is quite convenient because the approach is simple and easy to implement. However, the simplicity of the approach prevents the formulation of more elaborate querying tasks. For instance, queries which refer to both the structure and the content of the text cannot be formulated. To overcome this deficiency, more sophisticated query languages are required. Chapter 4 discusses various types of *query languages.* Since now the user might refer to the structure of a document in his query, this structure has to be defined. This is done by embedding the description of a document content and of its structure in a text language such as the Standard Generalized Markup Language (SGML). As illustrated in Figure 1.5, Chapter 6 is dedicated to the discussion of *text languages.*

Retrieval based on keywords might be of fairly low quality. Two possible reasons are as follows. First, the user query might be composed of too few terms which usually implies that the query context is poorly characterized. This is frequently the case, for instance, in the Web. This problem is dealt with through transformations in the query such as query expansion and user relevance feedback. Such *query operations* are covered in Chapter 5. Second, the set of keywords generated for a given document might fail to summarize its semantic content properly. This problem is dealt with through transformations in the text such as identification of noun groups to be used as keywords, stemming, and the use of a thesaurus. Additionally, for reasons of efficiency, text compression can be employed. Chapter 7 is dedicated to *text operations.*

Given the user query, the information system has to retrieve the documents which are related to that query. The potentially large size of the document collection (e.g., the Web is composed of millions of documents) implies that specialized indexing techniques must be used if efficient retrieval is to be achieved. Thus, to speed up the task of matching documents to queries, proper *indexing* and *search-*

ing techniques are used as discussed in Chapter 8. Additionally, query processing can be further accelerated through the adoption of *parallel and distributed IR* techniques as discussed in Chapter 9.

As illustrated in Figure 1.5, all the key issues regarding Text IR, from modeling to fast query processing, are covered in this book.

Modern user interfaces implement strategies which assist the user to form a query. The main objective is to allow him to define more precisely the context associated to his information need. The importance of query contextualization is a consequence of the difficulty normally faced by users during the querying process. Consider, for instance, the problem of quickly finding useful information in the Web. Navigation in hyperspace is not a good solution due to the absence of a logical and semantically well defined structure (the Web has no underlying logical model). A popular approach for specifying a user query in the Web consists of providing a set of keywords which are searched for. Unfortunately, the number of terms provided by a common user is small (typically, fewer than four) which usually implies that the query is vague. This means that new user interface paradigms which assist the user with the query formation process are required. Further, since a vague user query usually retrieves hundreds of documents, the conventional approach of displaying these documents as items of a scrolling list is clearly inadequate. To deal with this problem, new data visualization paradigms have been proposed in recent years. The main trend is towards visualization of a large subset of the retrieved documents at once and direct manipulation of the whole subset. *User interfaces* for assisting the user to form his query and current approaches for *visualization* of large data sets are covered in Chapter 10.

Following this, we discuss the application of IR techniques to multimedia data. The key issue is how to model, index, and search structured documents which contain multimedia objects such as digitized voice, images, and other binary data. *Models and query languages* for office and medical information retrieval systems are covered in Chapter 11. *Efficient indexing and searching* of multimedia objects is covered in Chapter 12. Some readers may argue that the models and techniques for multimedia retrieval are rather different from those for classic text retrieval. However, we take into account that images and text are usually together and that with the Web, other media types (such as video and audio) can also be mixed in. Therefore, we believe that in the future, all the above will be treated in a unified and consistent manner. Our book is a first step in that direction.

The final three chapters of the book are dedicated to applications of modern information retrieval: the Web, bibliographic systems, and digital libraries. As illustrated in Figure 1.5, Chapter 13 presents the Web and discusses the main problems related to the issue of searching the Web for useful information. Also, our discussion covers briefly the most popular search engines in the Web presenting particularities of their organization. Chapter 14 covers commercial document databases and online public access catalogs. Commercial document databases are still the largest information retrieval systems nowadays. LEXIS-NEXIS, for instance, has a database with 1.3 billion documents and attends to over 120 million query requests annually. Finally, Chapter 15 discusses modern digital

libraries. Architectural issues, models, prototypes, and standards are all covered. The discussion also introduces the '5S' model (streams, structures, spaces, scenarios and societies) as a framework for providing theoretical and practical unification of digital libraries.

1.6 How to Use this Book

Although several people have contributed chapters for this book, it is really a textbook. The contents and the structure of the book have been carefully designed by the two main authors who also authored or coauthored nine of the 15 chapters in the book. Further, all the contributed chapters have been judiciously edited and integrated into a unifying framework that provides uniformity in structure and style, a common glossary, a common bibliography, and appropriate cross-references. At the end of each chapter, a discussion on research issues, trends, and selected bibliography is included. This discussion should be useful for graduate students as well as for researchers. Furthermore, the book is complemented by a Web page with additional information and resources.

1.6.1 Teaching Suggestions

This textbook can be used in many different areas including computer science (CS), information systems, and library science. The following list gives suggested contents for different courses at the undergraduate and graduate level, based on syllabuses of many universities around the world:

- **Information Retrieval** (Computer Science, undergraduate): this is the standard course for many CS programs. The minimum content should include Chapters 1 to 8 and Chapter 10, that is, most of the part on Text IR complemented with the chapter on user interfaces. Some specific topics of those chapters, such as more advanced models for IR and sophisticated algorithms for indexing and searching, can be omitted to fit a one term course. The chapters on Applications of IR can be mentioned briefly at the end.
- **Advanced Information Retrieval** (Computer Science, graduate): similar to the previous course but with more detailed coverage of the various chapters particularly modeling and searching (assuming the previous course as a requirement). In addition, Chapter 9 and Chapters 13 to 15 should be covered completely. Emphasis on research problems and new results is a must.
- **Information Retrieval** (Information Systems, undergraduate): this course is similar to the CS course, but with a different emphasis. It should include Chapters 1 to 7 and Chapter 10. Some notions from Chapter 8 are

useful but not crucial. At the end, the system-oriented parts of the chapters on Applications of IR, in particular those on Bibliographic Systems and Digital Libraries, must be covered (this material can be complemented with topics from [501]).

- **Information Retrieval** (Library Science, undergraduate): similar to the previous course, but removing the more technical and advanced material of Chapters 2, 5, and 7. Also, greater emphasis should be put on the chapters on Bibliographic Systems and Digital Libraries. The course should be complemented with a thorough discussion of the user-centered view of the IR problem (for example, using the book by Allen [13]).
- **Multimedia Retrieval** (Computer Science, undergraduate or graduate): this course should include Chapters 1 to 3, 6, and 11 to 15. The emphasis could be on multimedia itself or on the integration of classical IR with multimedia. The course can be complemented with one of the many books on this topic, which are usually more broad and technical.
- **Topics in IR** (Computer Science, graduate): many chapters of the book can be used for this course. It can emphasize modeling and evaluation or user interfaces and visualization. It can also be focused on algorithms and data structures (in that case, [275] and [825] are good complements). A multimedia focus is also possible, starting with Chapters 11 and 12 and using more specific books later on.
- **Topics in IR** (Information Systems or Library Science, graduate) similar to the above but with emphasis on non-technical parts. For example, the course could cover modeling and evaluation, query languages, user interfaces, and visualization. The chapters on applications can also be considered.
- **Web Retrieval and Information Access** (generic, undergraduate or graduate): this course should emphasize hypertext, concepts coming from networks and distributed systems and multimedia. The kernel should be the basic models of Chapter 2 followed by Chapters 3, 4, and 6. Also, Chapters 11 and 13 to 15 should be discussed.
- **Digital Libraries** (generic, undergraduate or graduate): This course could start with part of Chapters 2 to 4 and 6, followed by Chapters 10, 14, and 15. The kernel of the course could be based on the book by Lesk [501].

More bibliography useful for many of the courses above is discussed in the last section of this chapter.

1.6.2 The Book's Web Page

As IR is a very dynamic area nowadays, a book by itself is not enough. For that reason (and many others), the book has a Web home page located and mirrored in the following places (mirrors in USA and Europe are also planned):

- Brazil: `http://www.dcc.ufmg.br/irbook`
- Chile: `http://sunsite.dcc.uchile.cl/irbook`

Comments, suggestions, contributions, or mistakes found are welcome through email to the contact authors given on the Web page.

The Web page contains the Table of Contents, Preface, Acknowledgements, Introduction, Glossary, and other appendices to the book. It also includes exercises and teaching materials that will be increasing in volume and changing with time. In addition, a reference collection (containing 1239 documents on Cystic Fibrosis and 100 information requests with extensive relevance evaluation [721]) is available for experimental purposes. Furthermore, the page includes useful pointers to IR syllabuses in different universities, IR research groups, IR publications, and other resources related to IR and this book. Finally, any new important results or additions to the book as well as an errata will be made publicly available there.

1.7 Bibliographic Discussion

Many other books have been written on information retrieval, and due to the current widespread interest in the subject, new books have appeared recently. In the following, we briefly compare our book with these previously published works.

Classic references in the field of information retrieval are the books by van Rijsbergen [785] and Salton and McGill [698]. Our distinction between data and information retrieval is borrowed from the former. Our definition of the information retrieval process is influenced by the latter. However, almost 20 years later, both books are now outdated and do not cover many of the new developments in information retrieval.

Three more recent and also well known references in information retrieval are the book edited by Frakes and Baeza-Yates [275], the book by Witten, Moffat, and Bell [825], and the book by Lesk [501]. All these three books are complementary to this book. The first is focused on data structures and algorithms for information retrieval and is useful whenever quick prototyping of a known algorithm is desired. The second is focused on indexing and compression, and covers images besides text. For instance, our definition of a textual image is borrowed from it. The third is focused on digital libraries and practical issues such as history, distribution, usability, economics, and property rights. On the issue of computer-centered and user-centered retrieval, a generic book on information systems that takes the latter view is due to Allen [13].

There are other complementary books for specific chapters. For example, there are many books on IR and hypertext. The same is true for generic or specific multimedia retrieval, as images, audio or video. Although not an information retrieval title, the book by Rosenfeld and Morville [682] on information architecture of the Web, is a good complement to our chapter on searching the

Web. The book by Menasce and Almeida [554] demonstrates how to use queueing theory for predicting Web server performance. In addition, there are many books that explain how to find information on the Web and how to use search engines.

The reference edited by Sparck Jones and Willet [414], which was long awaited, is really a collection of papers rather than an edited book. The coherence and breadth of coverage in our book makes it more appropriate as a textbook in a formal discipline. Nevertheless, this collection is a valuable research tool. A collection of papers on cross-language information retrieval was recently edited by Grefenstette [323]. This book is a good complement to ours for people interested in this particular topic. Additionally, a collection focused on intelligent IR was edited recently by Maybury [550], and another collection on natural language IR edited by Strzalkowski will appear soon [748].

The book by Korfhage [451] covers a lot less material and its coverage is not as detailed as ours. For instance, it includes no detailed discussion of digital libraries, the Web, multimedia, or parallel processing. Similarly, the books by Kowalski [459] and Shapiro *et al.* [719] do not cover these topics in detail, and have a different orientation. Finally, the recent book by Grossman and Frieder [326] does not discuss the Web, digital libraries, or visual interfaces.

For people interested in research results, the main journals on IR are: *Journal of the American Society of Information Sciences* (*JASIS*) published by Wiley and Sons, *ACM Transactions on Information Systems*, *Information Processing & Management* (IP&M, Elsevier), *Information Systems* (Elsevier), *Information Retrieval* (Kluwer), and *Knowledge and Information Systems* (Springer). The main conferences are: ACM SIGIR International Conference on Information Retrieval, ACM International Conference on Digital Libraries (ACM DL), ACM Conference on Information Knowledge and Management (CIKM), and Text REtrieval Conference (TREC). Regarding events of regional influence, we would like to acknowledge the SPIRE (South American Symposium on String Processing and Information Retrieval) symposium.

Chapter 2
Modeling

2.1 Introduction

Traditional information retrieval systems usually adopt index terms to index and retrieve documents. In a restricted sense, an index term is a keyword (or group of related words) which has some meaning of its own (i.e., which usually has the semantics of a noun). In its more general form, an index term is simply any word which appears in the text of a document in the collection. Retrieval based on index terms is simple but raises key questions regarding the information retrieval task. For instance, retrieval using index terms adopts as a fundamental foundation the idea that the semantics of the documents and of the user information need can be naturally expressed through sets of index terms. Clearly, this is a considerable oversimplification of the problem because a lot of the semantics in a document or user request is lost when we replace its text with a set of words. Furthermore, matching between each document and the user request is attempted in this very imprecise space of index terms. Thus, it is no surprise that the documents retrieved in response to a user request expressed as a set of keywords are frequently irrelevant. If one also considers that most users have no training in properly forming their queries, the problem is worsened with potentially disastrous results. The frequent dissatisfaction of Web users with the answers they normally obtain is just one good example of this fact.

Clearly, one central problem regarding information retrieval systems is the issue of predicting which documents are relevant and which are not. Such a decision is usually dependent on a ranking algorithm which attempts to establish a simple ordering of the documents retrieved. Documents appearing at the top of this ordering are considered to be more likely to be relevant. Thus, ranking algorithms are at the core of information retrieval systems.

A ranking algorithm operates according to basic premises regarding the notion of document relevance. Distinct sets of premises (regarding document relevance) yield distinct information retrieval models. The IR model adopted determines the predictions of what is relevant and what is not (i.e., the notion of relevance implemented by the system). The purpose of this chapter is to cover the most important information retrieval models proposed over the years. By

doing so, the chapter also provides a conceptual basis for most of the remaining chapters in this book.

We first propose a taxonomy for categorizing the 15 IR models we cover. Second, we distinguish between two types of user retrieval tasks: ad hoc and filtering. Third, we present a formal characterization of IR models which is useful for distinguishing the various components of a particular model. Last, we discuss each of the IR models included in our taxonomy.

2.2 A Taxonomy of Information Retrieval Models

The three classic models in information retrieval are called Boolean, vector, and probabilistic. In the Boolean model, documents and queries are represented as sets of index terms. Thus, as suggested in [327], we say that the model is *set theoretic*. In the vector model, documents and queries are represented as vectors in a t-dimensional space. Thus, we say that the model is *algebraic*. In the probabilistic model, the framework for modeling document and query representations is based on probability theory. Thus, as the name indicates, we say that the model is *probabilistic*.

Over the years, alternative modeling paradigms for each type of classic model (i.e., set theoretic, algebraic, and probabilistic) have been proposed. Regarding alternative set theoretic models, we distinguish the fuzzy and the extended Boolean models. Regarding alternative algebraic models, we distinguish the generalized vector, the latent semantic indexing, and the neural network models. Regarding alternative probabilistic models, we distinguish the inference network and the belief network models. Figure 2.1 illustrates a taxonomy of these information retrieval models.

Besides references to the text content, the model might also allow references to the structure normally present in written text. In this case, we say that we have a structured model. We distinguish two models for structured text retrieval namely, the non-overlapping lists model and the proximal nodes model.

As discussed in Chapter 1, the user task might be one of *browsing* (instead of retrieval). In Figure 2.1, we distinguish three models for browsing namely, flat, structure guided, and hypertext.

The organization of this chapter follows the taxonomy of information retrieval models depicted in the figure.We first discuss the three classic models. Second, we discuss the alternative models for each type of classic model. Third, we cover structured text retrieval models. At the end, we discuss models for browsing.

We emphasize that the IR model (Boolean, vector, probabilistic, etc.), the logical view of the documents (full text, set of index terms, etc.), and the user task (retrieval, browsing) are orthogonal aspects of a retrieval system as detailed in Chapter 1. Thus, despite the fact that some models are more appropriate for a certain user task than for another, the same IR model can be used with distinct document logical views to perform different user tasks. Figure 2.2 illustrates the retrieval models most frequently associated with each one of six distinct combinations of a document logical view and a user task.

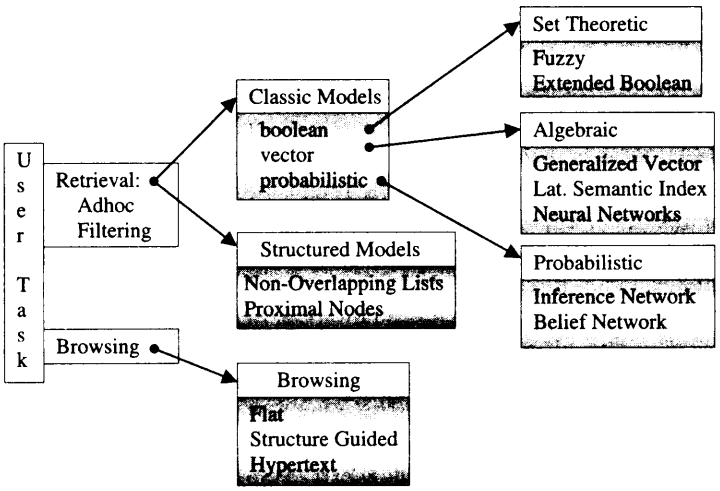

Figure 2.1 A taxonomy of information retrieval models.

LOGICAL VIEW OF DOCUMENTS

USER TASK	Index Terms	Full Text	Full Text + Structure
Retrieval	Classic Set Theoretic Algebraic Probabilistic	Classic Set Theoretic Algebraic Probabilistic	Structured
Browsing	Flat	Flat Hypertext	Structure Guided Hypertext

Figure 2.2 Retrieval models most frequently associated with distinct combinations of a document logical view and a user task.

2.3 Retrieval: Ad hoc and Filtering

In a conventional information retrieval system, the documents in the collection remain relatively static while new queries are submitted to the system. This operational mode has been termed *ad hoc* retrieval in recent years and is the

most common form of user task. A similar but distinct task is one in which the queries remain relatively static while new documents come into the system (and leave). For instance, this is the case with the stock market and with news wiring services. This operational mode has been termed *filtering*.

In a filtering task [74], a *user profile* describing the user's preferences is constructed. Such a profile is then compared to the incoming documents in an attempt to determine those which might be of interest to this particular user. For instance, this approach can be used to select a news article among thousands of articles which are broadcast each day. Other potential scenarios for the application of filtering include the selection of preferred judicial decisions, or the selection of articles from daily newspapers, etc.

Typically, the filtering task simply indicates to the user the documents which might be of interest to him. The task of determining which ones are really relevant is fully reserved to the user. Not even a ranking of the filtered documents is provided. A variation of this procedure is to rank the filtered documents and show this ranking to the user. The motivation is that the user can examine a smaller number of documents if he assumes that the ones at the top of this ranking are more likely to be relevant. This variation of filtering is called *routing* (see Chapter 3) but it is not popular.

Even if no ranking is presented to the user, the filtering task can compute an internal ranking to determine potentially relevant documents. For instance, documents with a ranking above a given threshold could be selected; the others would be discarded. Any IR model can be adopted to rank the documents, but the vector model is usually preferred due to its simplicity. At this point, we observe that filtering is really a type of user task (or operational mode) and not a model of information retrieval. Thus, the task of filtering and the IR model adopted are orthogonal aspects of an IR system.

In a filtering task, the crucial step is not the ranking itself but the construction of a user profile which truly reflects the user's preferences. Many approaches for constructing user profiles have been proposed and here we briefly discuss a couple of them.

A simplistic approach for constructing a user profile is to describe the profile through a set of keywords and to require the user to provide the necessary keywords. The approach is simplistic because it requires the user to do too much. In fact, if the user is not familiar with the service which generates the upcoming documents, he might find it fairly difficult to provide the keywords which appropriately describe his preferences in that context. Furthermore, an attempt by the user to familiarize himself with the vocabulary of the upcoming documents might turn into a tedious and time consuming exercise. Thus, despite its feasibility, requiring the user to precisely describe his profile might be impractical. A more elaborate alternative is to collect information from the user about his preferences and to use this information to build the user profile dynamically. This can be accomplished as follows.

In the very beginning, the user provides a set of keywords which describe an initial (and primitive) profile of his preferences. As new documents arrive, the system uses this profile to select documents which are potentially of interest and

shows them to the user. The user then goes through a relevance feedback cycle (see Chapter 5) in which he indicates not only the documents which are really relevant but also the documents which are non-relevant. The system uses this information to adjust the user profile description such that it reflects the new preferences just declared. Of course, with this procedure the profile is continually changing. Hopefully, however, it stabilizes after a while and no longer changes drastically (unless, of course, the user's interests shift suddenly). Chapter 5 illustrates mechanisms which can be used to dynamically update a keyword-based profile.

From the above, it should be clear that the filtering task can be viewed as a conventional information retrieval task in which the documents are the ones which keep arriving at the system. Ranking can be computed as before. The difficulty with filtering resides in describing appropriately the user's preferences in a user profile. The most common approaches for deriving a user profile are based on collecting relevant information from the user, deriving preferences from this information, and modifying the user profile accordingly. Since the number of potential applications of filtering keeps increasing, we should see in the future a renewed interest in the study and usage of the technique.

2.4 A Formal Characterization of IR Models

We have argued that the fundamental premises which form the basis for a ranking algorithm determine the IR model. Throughout this chapter, we will discuss different sets of such premises. However, before doing so, we should state clearly what exactly an IR model is. Our characterization is as follows.

Definition *An information retrieval* model *is a quadruple* $[\mathbf{D}, \mathbf{Q}, \mathcal{F}, R(q_i, d_j)]$ *where*

(1) $\mathbf{D}$ *is a set composed of logical views (or representations) for the documents in the collection.*

(2) $\mathbf{Q}$ *is a set composed of logical views (or representations) for the user information needs. Such representations are called* queries.

(3) $\mathcal{F}$ *is a framework for modeling document representations, queries, and their relationships.*

(4) $R(q_i, d_j)$ *is a ranking function which associates a real number with a query* $q_i \in \mathbf{Q}$ *and a document representation* $d_j \in \mathbf{D}$. *Such ranking defines an ordering among the documents with regard to the query* q_i.

To build a model, we think first of representations for the documents and for the user information need. Given these representations, we then conceive the framework in which they can be modeled. This framework should also provide the intuition for constructing a ranking function. For instance, for the classic Boolean model, the framework is composed of sets of documents and the standard operations on sets. For the classic vector model, the framework is composed of a

t-dimensional vectorial space and standard linear algebra operations on vectors. For the classic probabilistic model, the framework is composed of sets, standard probability operations, and the Bayes' theorem.

In the remainder of this chapter, we discuss the various IR models shown in Figure 2.1. Throughout the discussion, we do not explicitly instantiate the components $\mathbf{D}$, $\mathbf{Q}$, $\mathcal{F}$, and $R(q_i, d_j)$ of each model. Such components should be quite clear from the discussion and can be easily inferred.

2.5 Classic Information Retrieval

In this section we briefly present the three classic models in information retrieval namely, the Boolean, the vector, and the probabilistic models.

2.5.1 Basic Concepts

The classic models in information retrieval consider that each document is described by a set of representative keywords called index terms. An *index term* is simply a (document) word whose semantics helps in remembering the document's main themes. Thus, index terms are used to index and summarize the document contents. In general, index terms are mainly nouns because nouns have meaning by themselves and thus, their semantics is easier to identify and to grasp. Adjectives, adverbs, and connectives are less useful as index terms because they work mainly as complements. However, it might be interesting to consider all the distinct words in a document collection as index terms. For instance, this approach is adopted by some Web search engines as discussed in Chapter 13 (in which case, the document logical view is *full text*). We postpone a discussion on the problem of how to generate index terms until Chapter 7, where the issue is covered in detail.

Given a set of index terms for a document, we notice that not all terms are equally useful for describing the document contents. In fact, there are index terms which are simply vaguer than others. Deciding on the importance of a term for summarizing the contents of a document is not a trivial issue. Despite this difficulty, there are properties of an index term which are easily measured and which are useful for evaluating the potential of a term as such. For instance, consider a collection with a hundred thousand documents. A word which appears in each of the one hundred thousand documents is completely useless as an index term because it does not tell us anything about which documents the user might be interested in. On the other hand, a word which appears in just five documents is quite useful because it narrows down considerably the space of documents which might be of interest to the user. Thus, it should be clear that distinct index terms have varying relevance when used to describe document contents. This effect is captured through the assignment of numerical *weights* to each index term of a document.