Abstract

Title of Dissertation: Adaptive Vector Space Text Filtering
for Monolingual and
Cross-Language Applications

Douglas William Oard, Doctor of Philosophy, 1996

Dissertation directed by: Professor Nicholas DeClaris
Department of Electrical Engineering, UMCP
Department of Pathology, UMAB

Adaptive Multilingual Text Filtering is an innovative approach for enhancing
the usability of the global information infrastructure. With the recent explosive
growth of intranets and the Internet, mediated access to networked information
is becoming an increasingly important problem. Much of this information is
in (or can be converted to) text form, and an increasing amount is expressed
in languages other than English. Text filtering is an information access pro-
cess in which dynamic information sources are matched against relatively stable
information needs; adaptive text filtering systems seek to assist users by auto-
matically improving their profile of the users' information needs over time; and
adaptive multilingual text filtering systems seek to do so for collections of text
which contain more than one language. The distinguishing feature of adaptive

multilingual text filtering is that evidence obtained by observing reading behavior on text in one language can be used to predict interest in texts in another language. Existing multilingual text filtering systems are not adaptive, requiring instead that users explicitly specify their information needs using words from each language of interest.

In this dissertation, present practice in adaptive text filtering is reviewed and a simple monolingual technique based on Latent Semantic Indexing with obvious potential for application to multilingual filtering is selected. Further refinement of this monolingual Latent Semantic Indexing filtering technique is found to result in no performance improvement, so that technique is adopted unchanged as the basis for multilingual filtering experiments. The present state of the art in the related multilingual text retrieval problem is then surveyed. Cross-language application of Latent Semantic Indexing is described and a second technique based on fully automatic machine translation is introduced. A "vector translation" technique designed specifically for adaptive multilingual text filtering is then introduced and an experimental methodology which exploits existing test collections is presented. Experimental results reveal a cost-performance tradeoff among the techniques. Implications of these results for future work on adaptive multilingual text filtering are discussed, and the dissertation concludes with an assessment of the present state of the field and the potential for further advances.

Adaptive Vector Space Text Filtering
for Monolingual and
Cross-Language Applications

by

Douglas William Oard

Dissertation submitted to the Faculty of the Graduate School
of the University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1996

Advisory Committee:

Professor Nicholas DeClaris, Chairman/Advisor
Professor Bonnie Dorr, Coadvisor
Professor Gary Marchionini
Professor Robert Newcomb
Professor Charles Silio

Dedication

To my parents, without whom this would not have been possible.

Acknowledgements

A large number of people have contributed to my success in this endeavor. Dr Nicholas DeClaris, Dr. Bonnie Dorr, Dr. Christos Faloutsos and Dr. Gary Marchionini have all helped to shape the way that I think about information filtering, both as a process and as a task which can be automated. For their inspiration and the flashes of insight that we shared I am indebted to all of them. It is unfortunate that summer travel schedules made it impossible for Dr. Christos Faloutsos to participate in the dissertation committee. He certainly was there in spirit.

A number of other people have made important contributions as well. The generous support of the Logos Corporation, and in particular Dr. Scott Bennett, made the text translation experiment possible. Without Wade Shen's tireless efforts to perform the statistical word alignments that serve as a basis for the vector translation technique it

Table of Contents

List of Tables

List of Figures

Adaptive Vector Space Text Filtering
for Monolingual and
Cross-Language Applications

Douglas William Oard

August 2, 1996

Revision history:

Version 1 — June 23 — to Dr. DeClaris for overseas travel

Version 2 — June 25 — to Dr. Dorr for review

Version 3 — July 5 — to Dr. Newcomb and Dr. Marchionini for review

Version 4 — July 9 — Final version to committee

Version 5 — July 25 — Incorporating all comments except Dr. DeClaris'

Version 6 — July 30 — Proofreading version for Graduate School

Version 7 — July 31 — Verification version for Dr. DeClaris

Version 8 — August 1 — Final Version

This version is complete with respect to the content and organization, and it conforms completely to the Graduate School's format guidelines. All requested changes are incorporated.

The final version of the dissertation must be turned in to the Graduate office by Monday August 5. I must leave town for three weeks on Friday night, August 2, however, so that is the practical deadline in my case.

This comment page is not part of the dissertation.

Typeset by LaTeX using the `dissertation` style by Pablo A. Straub, University of Maryland.

Chapter 1

Introduction

With the recent explosive growth of the Internet and other sources of networked information, automatic mediation of access to networked information sources is becoming an increasingly important problem. Much of this information is in (or can be converted to) text form, and an increasing amount is expressed in languages other than English. This dissertation investigates a specific approach to managing access to information, adaptive text filtering, and it presents the first application of adaptive text filtering to document collections which contain more than one language.

Information filtering is an information access process in which the information sources are matched against long-term information needs. Text filtering systems match representations of text that are constructed from the information sources with representations of relatively stable information needs and display the results in ways designed to help users select useful information sources. By "text" we mean sequential collections of codes which represent characters that are organized into entities which we call "documents" to convey information using human language. Some typical sources of text on the Internet are elec-

tronic mail, World Wide Web pages, and newswire articles, but text filtering techniques can be applied to a far larger range of sources. Optical character recognition from images of printed documents and speech recognition technology, for example, produce (sometimes incorrect) text representations that are derived from other media. In this dissertation, however, we are not concerned with how the text was acquired, but rather in how it is used.

We are interested in "adaptive" text filtering systems in particular because they seek to assist human users by modeling some aspects of human cognition. Adaptive text filtering systems are text filtering systems which strive to automatically improve the accuracy of their representation of the user's interest over time. Adaptive text filtering is a dynamic research area, and monolingual adaptive text filtering systems are presently receiving substantial attention because they offer the prospect of matching many of the available sources of information with existing information needs [105].

In this dissertation we describe the adaptive text filtering process in detail and report the results of experiments investigating two important aspects of adaptive text filtering. Since we seek to model some aspects of human cognition, the utility of the adaptive text filtering systems we produce will depend critically on the ability of those systems to develop representations of the users' information needs and to use those representations to help the user to identify newly arrived information that he or she will find useful. Thus, our first investigation sought to determine whether an existing adaptive text filtering system could be extended to achieve improved performance. In Chapters 2 and 3 we survey the known techniques for text filtering, define performance measures, describe in detail the technique we have chosen to extend, motivate and present our extension, describe

the design of an experiment to evaluate the new technique, present the results of that experiment, and discuss the implications of those results.

The results of this first experiment reveal that our new technique (which we call the "Gaussian User Model") performs no better than the existing technique (which we refer to below as the "LSI-mean" technique) that we had sought to extend, despite the use of a more sophisticated information need representation in the Gaussian User Model. The LSI-mean technique does, however, provide an excellent basis for a second set of experiments on an important new application of adaptive text filtering techniques, what we call "adaptive multilingual text filtering."

In the second half of the dissertation we present the results of what we believe is the first experimental investigation of adaptive multilingual text filtering. Adaptive multilingual text filtering systems are adaptive text filtering systems which are able to exploit collections of text that contain more than one language [34, 102]. This general formulation includes two special cases:

- Each document is written in a single language, but more than one language is included in the collection.

- Some documents in the collection contain more than one language.

In Chapters 4 and 5 we survey related techniques from the field known as "multilingual text retrieval," describe three adaptive multilingual text filtering techniques that we have developed, present the results of an experimental evaluation of their relative performance, and identify the important issues for further research on this topic that our work has uncovered. Section 1.3 describes the coverage of each aspect of our investigation by chapter.

Our work on adaptive multilingual text filtering is built on the foundation provided by our experiments comparing the performance of the Gaussian User Model with the LSI-mean technique. Readers with a particular interest in adaptive multilingual text filtering will thus find it beneficial to read the entire dissertation in the order presented, and the introductory and concluding remarks in each chapter are written with that reader in mind. Readers wishing to focus on adaptive text filtering more generally may wish to read the first half in detail and then skim the remainder since the vast majority of the details we consider in the second half of the dissertation are unique to multilingual applications.

1.1   Background

Adaptive multilingual text filtering systems seek to help information consumers identify information sources which satisfy relatively stable information needs by adapting their behavior over time to conform more closely with their user's information seeking behavior, and doing so without regard to the language in which the information is expressed. Among the potential applications for adaptive multilingual text filtering are:

- A medical doctor specializing in the diagnosis and treatment of a specific type of medical problem seeks to be informed of patients receiving care from any practitioner in an international health care network when the information added to the patient's record is similar to notations that have often been seen in interesting cases. Doctors participating in the network generally make record entries in their native language, but the specialist is able to read medical record entries in many languages if they are germane

to their interests.

- A large international engineering project exchanges documents electronically among participating firms and government agencies. Documents must be routed to engineers with cognizance over specific functions, even when the author of a document lacks sufficient insight into the project's organizational structure to identify appropriate recipients. Documents may be in one of several languages, so translation may be required before the documents will be useful to individual engineers.

- A research scientist wishes to be informed of scientific papers published on a topic, but those papers appear in a broad variety of journals and conference proceedings and in a large number of languages. Most of the papers include bibliographic citations to related work published in English, making it possible to gain some familiarity with the nature of the reported work without being able to read the language of publication.

- A news bureau seeks to monitor worldwide television broadcasts in order to identify news coverage of specific types of events. Selection of material is based on speech to text conversion, augmented, where available, by closed caption text annotations. Even when the audio and closed captions are a language not understood by the news bureau staff, the staff is often able to make use of the video content of the selected programs.

- A commodities trader seeks timely information from newswire sources on issues which affect the price of a specific commodity. The trader's information needs are fairly specific, change little over time, and may be satisfied by news articles in any of several languages which the trader is able to

read.

- The headquarters of a multinational corporation receives clippings of articles and advertisements appearing in the trade and popular press by fax from offices in many countries. These documents must be routed to the appropriate individuals based on text extracted from them using optical character recognition. The text may be in any of several languages that are commonly used by the media in the countries where the corporation does business. Individual recipients may elect to submit some of these documents for translation if they appear interesting and the recipient is unable to make use of them in their original language.

All of these applications share four important features which together define the multilingual text filtering problem:

- Text extracted from the available information sources provides a basis for identification of potentially useful information.

- That text may be in any of several languages, perhaps even including more than one language within a single document.

- The information need is fairly stable and specific, making it possible to acquire and exploit evidence about that need over time.

- Users can potentially benefit from automatic assistance with the development of a suitable representations of their information needs.

It is important to note that in many of these examples it suffices to detect useful documents. Translation of documents which the user wishes to examine

6

is sometimes a useful adjunct to adaptive multilingual text filtering, but it is not a central part of the problem which we wish to solve.

The use of text analysis to detect useful information is not a new problem [89]. A wide variety of techniques have been developed for this purpose in a field known as "information retrieval." Some of these techniques are suitable for selection of information in multiple languages, a specialization sometimes referred to as "multilingual text retrieval" [104]. Another area of specialized study, what we have called "adaptive text filtering" is the design of systems that are able to refine their representation of relatively stable information needs based on observations of the user's behavior [105].

Adaptive text filtering draws from two established lines of research, text retrieval and user modeling. Text retrieval research studies the closely related problem of how automated systems can help users retrieve texts from a relatively static collection, often in response to episodic requests that have little relation to those which came before or after [129]. User modeling studies how representations of some aspect of the user can be acquired and then used to help that user perform a task [124]. In both cases, representation of the objects to be manipulated is of central importance. Much of the present practice in text retrieval, and hence in text filtering as well, is based on noting the occurrence of terms in a document, rather than on the deeper semantic information conveyed by how the terms are combined.

A vector of term frequencies provides a convenient representation for these occurrence-based techniques. The vector representation is formed by counting the number of occurrences of each unique word in a document and storing that

value in one element of a list.[1] Figure 1.1 illustrates how vectors representing two different sentences can be constructed using the same set of words. In the first sentence, "the" occurs twice, so the corresponding value is 2. The entries for "child," "joy" and "trick" are assigned a value of 0 because they do not occur at all in that sentence. Longer documents will, of course, have more nonzero entries. But as additional documents are added to a collection, each will typically introduce several new words—words which will have a count of zero in all of the previous documents. So eventually the number of zero entries in each individual vector will be far larger than the number of nonzero entries.

back	brown	child	dog's	fox	joy	jumped	lazy	over	quick	the	trick
1	1	0	1	1	0	1	1	1	1	2	0

"The quick brown fox jumped over the lazy dog's back."

0	1	1	1	0	1	1	0	1	0	2	1

"The child jumped for joy over the brown dog's trick."
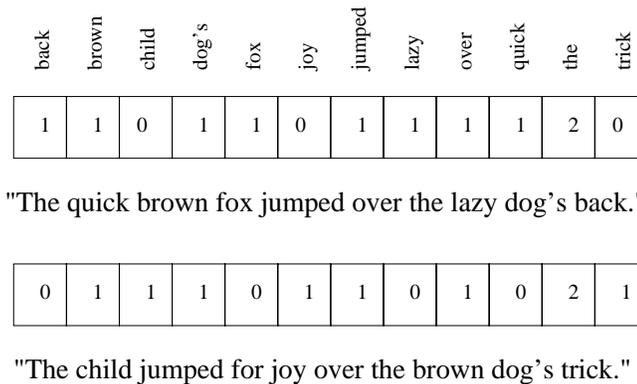
Figure 1.1: Vector representation of documents containing a single sentence.

There are several ways in which vector representations can be used to match documents with information needs. One simple approach, known as the "vector space method" is to treat each vector as if it specified the coordinates of the endpoint of a line, the other end of which is at the origin. The angle between the lines defined by two vectors is then used as an approximate measure of the similarity of the documents represented by those vectors. Figure 1.2 depicts this

---

[1]In Chapter 2 we describe how other indexing terms can be selected. We will consistently use individual words in the examples we present.

process using the two-dimensional vector space defined by values associated with "the" and "fox." Visualization of this angular comparison in high-dimensional spaces is more challenging, but the mathematical formulation of the angular difference that is presented in Chapter 2 is the same regardless of the number of dimensions.
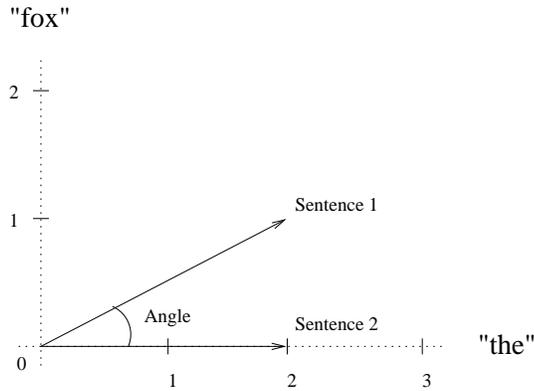
Figure 1.2: Computation of angular similarity in a vector space.

In the vector space method, one of the vectors is used to represent the user's information need, and all of the other vectors (and therefore all of the documents being considered) are sorted ("ranked") so that the most similar ones are at the top of the list. Chapter 2 describes two other ways of using vector representations to help select documents (probabilistic models and exact match selection), but all of the adaptive multilingual text filtering techniques developed in this dissertation have their roots in the vector space method.

Vector representations are quite simple, and that simplicity necessarily limits the performance of any document detection technique which is based solely on vector representations. But many practical applications (arguably including all of the above examples) require robust performance across a broad range of topics (what we shall refer to as "broad-domain" performance), and vector represen-

tations are by nature equally well suited to any domain. Furthermore, ranked output fosters a powerful synergy between the system and the user that permits the two together to detect useful documents which neither would be likely to reliably select alone. We shall have more to say on this point in Chapter 2. Finally, the performance of techniques based on vector representations can be enhanced significantly by replacing the raw term frequency values with "weights" that are computed using both individual vectors and summaries of information about language use that can be discerned from an entire collection of vectors. This issue is described briefly in Chapter 2 and then discussed in detail in Chapter 3.

In text retrieval applications of the vector space method, the vector which represents the information need is typically constructed directly from a set of terms provided by the user. Although the same approach can be used for information filtering, the persistence of the information need in the text filtering case makes it practical to apply user modeling techniques as well. We distinguish the case in which user models are constructed and exploited to support a text filtering process by referring to it as "adaptive" text filtering, the problem studied in this dissertation.

Vector space user models are those which use the locations of interesting vectors in high-dimensional "feature space" to identify those associated with useful documents. For example, Dumais has combined an enhancement to the basic vector space method (Latent Semantic Indexing or LSI) with a technique for computing a single vector which represents an information need to produce the LSI-mean technique we referred to above. An angular similarity measure like the one depicted in Figure 1.2 is then used to produce sorted lists in which three quarters of the top ranked documents are judged by humans to be useful [41].

1.2 Contributions

This dissertation advances the state of the art in the following four ways:

- It develops and demonstrates three adaptive multilingual text filtering techniques. This represents the first application of any multilingual technique to the adaptive filtering problem, an advance made possible by our development of a suitable evaluation methodology. Specific contributions include:

  - Applies Latent Semantic Indexing to adaptive multilingual text filtering, producing a technique that can be automatically trained for any domain and combination of languages by using existing collections of translated texts (Chapter 5).

  - Develops a "text translation" technique which can perform adaptive multilingual text filtering when no suitable collection of translated texts exists (Chapter 5).

  - Develops a novel vector translation technique for adaptive multilingual text filtering which exploits reinforcement effects that result from the interaction between the text representation and the adaptation technique (Chapter 5).

  - Develops an evaluation methodology which exploits existing test collections to measure the relative performance of adaptive multilingual text filtering techniques (Chapter 5).

  - Using that methodology to determine the relative performance of the three techniques (Chapter 5).

11

- It develops and evaluates a new technique for adaptive text filtering. Specific contributions include:

  - Extends the LSI-mean adaptive text filtering technique by incorporating a more sophisticated information need representation to produce the Gaussian User Model (Chapter 3).

  - Applies an existing evaluation methodology to compare the performance of the Gaussian User Model with that of the LSI-mean technique (Chapter 3).

  - Interprets the experimental results to identify fundamental limitations on the performance of adaptive text filtering techniques which exploit vector space methods (Chapter 3).

- It synthesizes a comprehensive perspective of the text filtering process, integrating diverse research to create a consistent definition of the process and automated systems which support that process (Chapter 2).

- It unifies two nearly disjoint bodies of literature on multilingual text retrieval that have developed in separate research communities, identifying commonalities and significant differences between the techniques which have been developed (Chapter 4).

1.3   Organization of the Dissertation

The ultimate goal towards which our presentation builds is the development and comparison of a representative set of multilingual techniques which are well suited to the adaptive text filtering problem. We have chosen this approach

because our multilingual techniques are built on the foundation established in the first part of the dissertation. The first part of the dissertation should also prove useful to anyone wishing to deepen their understanding of important issues in adaptive text filtering more generally.

We begin by surveying the present practice in text filtering in Chapter 2. Theoretical frameworks drawn from text retrieval, machine learning and user modeling are described, a wide variety of approaches are reviewed, and both experimental and operational systems are discussed. There are many open questions regarding the design of adaptive text filtering systems, so we have selected a promising technique based on the vector space method for further study.

In Chapter 3 we build on the LSI-mean technique, generalizing that technique to produce the an adaptive text filtering technique that refer to as the Gaussian User Model. The experimental results we present in that chapter reveal that the Gaussian User Model can not exceed the performance of the LSI-mean technique on applications for which relatively few positive examples are available for training. What makes the LSI-mean technique particularly interesting is that it has a natural application to adaptive multilingual text filtering, an adaptation we call "Latent Semantic Coindexing."

Although we are not aware of any prior work on adaptive multilingual text filtering, an exceptionally rich body of literature on multilingual text retrieval exists. We survey this work in Chapter 4 in order to identify approaches which could provide a basis for adaptive filtering applications. In addition to Latent Semantic Coindexing, we identify a technique based on fully automatic machine translation which has a straightforward application to adaptive multilingual text filtering that we call "Text Translation."

In Chapter 5 we describe the application of these two techniques to adaptive multilingual text retrieval and present the "Vector Translation" technique which we have developed specifically for adaptive multilingual text filtering. We present experimental results which indicate that all three techniques achieve useful performance levels. Although the nature of the available text collections prevents us from drawing definitive conclusions about the performance of each technique, we are able to identify relative differences which reveal a cost-performance tradeoff.

This dissertation represents the initial investigation of a rich research area with numerous practical applications. In Chapter 6 we identify a number of important topics that form a basis for further work in this area, the most important of which is the construction of better test collections.

We have demonstrated that adaptive multilingual text filtering is both possible and practical, and we have begun to identify the specific techniques which should be considered. Our evaluation methodology is best suited to determining whether documents judged to be useful by humans can be identified automatically, and it is on that portion of the problem that we have focused our efforts. Issues of efficiency and usability are addressed where appropriate, but the ability of the algorithms we have developed to identify useful documents under laboratory conditions is the principal focus of this first investigation of adaptive multilingual text filtering.

Chapter 2

Text Filtering

With the growth of the Internet and other networked information, research in automatic mediation of access to networked information has exploded in recent years [105]. This chapter reviews existing work on text filtering, a type of "information seeking."

2.1 Background

We use "information seeking" as an overarching term to describe any processes by which users seek to obtain information from automated information systems [92]. Table 2.1 shows common types of information seeking processes. In the "information filtering" process the user is assumed to be seeking information which addresses a specific long-term interest. In this chapter we introduce the information filtering problem in some detail, describe the specific techniques used for "text filtering," (the case in which the information sought is in text form), and identify the techniques that are well suited for multilingual applications.

Information filtering systems are typically designed to sort through large volumes of dynamically generated information and present the user with sources

Process	Information Need	Information Sources
Information Filtering	Stable & Specific	Dynamic & Unstructured
Information Retrieval	Dynamic & Specific	Stable & Unstructured
Database Access	Dynamic & Specific	Stable & Structured
Information Extraction	Specific	Unstructured
Alerting	Stable and Specific	Dynamic
Browsing	Broad	Unspecified
Entertainment	Unspecified	Unspecified

Table 2.1: Examples of information seeking processes.

of information that are likely to satisfy his or her information requirement. By "information sources" we mean entities which contain information in a form that can be interpreted by a user. We commonly refer to information sources which contain text as "documents," but in other contexts these sources may be audio, still or moving images, or even people. The information filtering system may either provide these entities directly (which is practical when the entities are easily replicated), or it may provide the user with references to the entities.

This description of information filtering leads immediately to three subtasks: collecting the information sources, selecting the information sources, and displaying the information sources. Figure 2.1 depicts this subdivision, one which is applicable to a wide variety of information seeking processes. The same three tasks are also fundamental to a process commonly referred to as "information retrieval" in which the system is presented with a query by the user and expected to produce information sources which the user finds useful. "Text retrieval," the specialization of information retrieval to retrieve text, has an extensive research heritage. In one of the classic works on information filtering, this observation led Belkin and Croft to suggest that the information filtering process would be an

attractive application for techniques that had already developed for information retrieval systems [5].

Collection	→	Selection	→	Display

Figure 2.1: Information seeking task diagram.

The distinction between process and system is fundamental to understanding the difference between information filtering and information retrieval. By "process" we mean an activity conducted by humans, perhaps with the assistance of a machine. When we refer to a type of "system" we mean an automated system (i.e., a machine) that is designed to support humans who are engaged in that process. So an information filtering system is a system that is intended by its designers to support an information filtering process. Much of the confusion that arises on this issue can be traced back to the creative application of techniques that were designed originally to support one type of information seeking process (e.g., information retrieval) to support a different type of information seeking process(e.g., information filtering).

Any information seeking process begins with the users' goals. The distinguishing features of the information filtering process are that the users' information needs (or "interests") are relatively specific (a point we shall come back to when we define browsing), and that those interests change relatively slowly with respect to the rate at which information sources become available. Although the information retrieval process is also restricted to specific information needs, historically information retrieval research has sought to develop systems which use relatively stable information sources to respond to collections of (possibly) unrelated queries. So a traditional information retrieval system can be used to

perform an information filtering process by repeatedly accumulating newly arrived documents for a short period, issuing an unchanging query against those documents, and then flushing the unselected documents. But the information filtering process is distinguished from the information retrieval process by the nature of the user's goal. Figure 2.2 depicts this distinction graphically. While the grand challenge for information seeking systems is to match rapidly changing information with highly variable interests, information retrieval and information filtering both explore important areas of this problem space for which a number of practical applications exist.
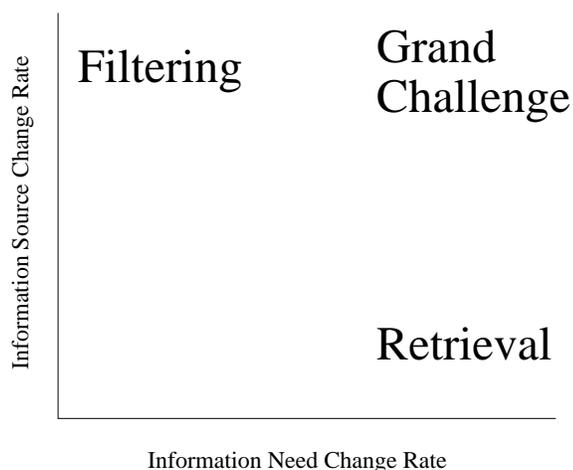
Figure 2.2: Information seeking processes for relatively specific information needs.

It is useful to highlight the distinction between information filtering and information retrieval because systems designed to support the information filtering process can exploit evidence about relatively stable interests to develop sophisticated models of the users' information needs. Information filtering can be viewed as an application of user modeling techniques to facilitate information seeking in dynamic environments. In summary, the design of information filtering systems

can be based on two established lines of research, information retrieval and user modeling.

2.1.1   Collection and Display

This chapter describes the design of systems to support the text filtering process with particular emphasis on the information selection component. Because such an emphasis might leave the reader with the mistaken impression that collection and display are lesser challenges, we pause briefly to describe the relationship between selection and the other two components depicted in figure 2.1.

Dynamic information can be collected actively (e.g., with autonomous agents over the World Wide Web), collected passively (e.g., from a newswire feed) or some combination of the two. Early descriptions of the information filtering problem implicitly assumed passive collection (c.f. [38, 65]). As the amount of electronically accessible information has exploded, active collection has become increasingly important (c.f. [159]). Active collection techniques can benefit from a close coupling between the collection and selection modules because they exploit both user and network models to perform information seeking actions in a network on behalf of the user. In a fully integrated information filtering system, some aspects of user model design are likely to be common to the two modules. That commonality would provide a basis for sharing information about user needs across the inter-module interface. But because the purpose of the collection module is to choose whether to obtain information before that information is known while the purpose of the selection module is to choose information to retain for display to the user once that information has been collected, the user model for the selection module is not likely to be identical to the user model

for the collection module. In the succeeding sections we will generally limit the discussion to systems which use passive collection techniques, both because this choice allows us to concentrate on the selection component and because there has been little reported on how the two components can be integrated.

Such a clean division is not possible for the interface between the selection and the display components, however. The goal of an information filtering system is to enhance the user's ability to identify useful information sources. While this can be accomplished by automatically choosing which sources of information to display, experience has shown that user satisfaction can be enhanced in interactive applications by using techniques which exploit the strengths of both humans and machines.

A personalized electronic conference system that lists submissions in order of decreasing likelihood of user interest is one example of such an approach. The automatic system can use computationally efficient techniques to place documents which are likely to be interesting near the top of the list, and then users can rapidly apply sophisticated heuristics (such as word sense interpretation and source authority evaluation) to select those documents most likely to meet their information need. If the system has produced a good rank ordering, the density of useful documents should be greatest near the top of the list. As the user proceeds down the list, selecting interesting documents to review, he or she should thus observe that the number of useful documents is decreasing. By allowing the human to adaptively choose to terminate their information seeking activity based in part on the observed density of useful documents, human and machine synergistically achieve better performance than either could achieve alone.

In other words, in interactive applications an imperfectly ranked list (re-

ferred to as "ranked output") can be superior to an imperfectly selected set of documents (referred to as "exact match" selection) because humans are able to adaptively choose the set size based on the same heuristics that they use to choose which documents to read. The choice of a ranked output display design imposes requirements on the selection module, however. Because the display module must rank the documents, the selection module must provide some basis (e.g., a numeric "status value") from which the ranking can be constructed. Display design is a rich research area in its own right, but our discussion of the issue is focused solely on aspects of the display design that impose requirements on the selection module.

2.1.2   Other Information Seeking Processes

We have already mentioned information retrieval, but there are other information seeking processes for which the decomposition in figure 2.1 is appropriate. One of the most familiar is the process of retrieving information from a database. The distinguishing feature of the database retrieval process is that the output will be information,[1] while in information filtering (or retrieval), the output is a set of entities (e.g., documents) which contain the information which is sought [11]. For example, using an library catalog to find the title of a book would be a database access process. Using the same system to discover whether any new books about a particular topic have been added to the collection would be an information filtering process. As this example shows, database systems can be

---

[1]While it is common to draw a distinction between information and data in which the concept of "information" includes some basis for its interpretation, our focus on selection makes it possible to combine the two concepts and refer to both as "information."

applied to information filtering processes, and we will present examples of this in section 2.4.

Another process that can be described using figure 2.1 is information extraction. The information extraction process is similar to database access in that the goal is to provide information to the user, rather than entities which contain information. It is distinguished from the database access process by the nature of the sources from which that information is obtained. In the database access process information is obtained from some type of database (e.g., a relational database), while in information extraction the information is less well structured (e.g., the body of an electronic mail message). Information extraction techniques are sometimes found in the selection module of a text filtering process, helping to represent texts in a way that facilitates selection.

One interesting variation on the information extraction and database access processes is what is commonly referred to as "alerting." In the alerting process the information need is assumed to be relatively stable with respect to the rate at which the information itself is changing.[2] Monitoring an electronic mailbox and alerting the user whenever mail from a specific user arrives is one example of an information alerting process. Presenting mail from that user first in a sorted list would be an example of information filtering.

The fields of database retrieval, information extraction, and alerting all contribute ideas and techniques to text filtering practice, and all three fields benefit from advances in text filtering research. We do not intend to comprehensively review those research areas, but we do occasionally mention how relevant tech-

---

[2]Recall that in an information filtering process it is the information sources, rather than the information itself, which change.

nologies developed to support those processes can be applied to support the information filtering process.

Finally, "browsing" is another information seeking process for which the decomposition shown in figure 2.1 is appropriate. Since browsing can be performed on either static or dynamic information sources, browsing has aspects similar to both information filtering and information retrieval. "Surfing the World Wide Web" is an example of browsing relatively static information, while reading an online newspaper would be an example of browsing dynamic information. The distinguishing feature of browsing is that the users' interests are assumed to be broader than in the information filtering or retrieval processes. Precisely what is meant by "broader" is difficult to define, however, and the distinction is often simply a matter of judgement. In order to sharpen the distinction for the purpose of this chapter, we propose an operational definition of browsing. When an interest is so broad that it cannot be represented effectively in an information filtering (or retrieval) system, we will refer to the information seeking process as browsing rather than as filtering or retrieval. In other words, we propose that researchers seek to characterize the broadest interests for which their information filtering systems are useful, and then refer to the limitations they discover in that way as the dividing line between filtering and browsing for their system.

2.2   Terminology

In a field as diverse as information filtering it is inevitable that a rich and sometimes conflicting set of terminology would emerge. Sometimes this is simply the result of differing perspectives, other times new terminology is needed to convey

subtly different meanings. For example, "information retrieval" is sometimes used expansively to include information filtering. But it is also commonly used in the more restricted sense that we have defined. Information filtering is alternatively referred to as "routing" (with a heritage in message processing) as "Selective Dissemination of Information" or "SDI" (with a heritage in library science), as "current awareness," and as "data mining." Sometimes routing is used to indicate that every document goes to some (and perhaps exactly one) user. Information filtering is sometimes associated with passive collection of information, and is sometimes meant to imply that an all-or-nothing (i.e., unranked) selection is required. SDI is sometimes used to imply that the profiles which describe the information need are constructed manually. The use of "current awareness" is sometimes meant to imply selection of new information based solely on the title of a journal, magazine, or other serial publication. And "data mining" is sometimes taken to imply that vast quantities of information are available simultaneously. All of those interpretations have a historical basis, but it is not uncommon to find these terms used to describe systems which lack the distinguishing characteristics of their historical antecedents. We shall avoid this problem by referring to all of these variations as "information filtering."

Taylor defined four types of information need (visceral, conscious, formalized, and compromised) that reflected the process of moving from the actual (but perhaps unrecognized) need for information to an expression of the need which could be represented in an information system [149]. In common use, however, application of the terminology is unfortunately not nearly so precise. The visceral information need is often referred to as an "interest" or simply as an "information need." But it is occasionally referred to as a topic, a term that is sometimes

(e.g., in the TREC evaluation we describe in section 2.4) used to describe the formalized (i.e., the human expression of) the information need. And in some experimental work, the visceral information need is referred to as a "query" even though "query" is the traditional term for Perry's concept of a compromised information need that could be submitted to an information retrieval system. In this chapter, we use "interest" and "information need" interchangeably to refer to the visceral information need, and reserve the use of the terms "topic" and "query" for their more specific meanings.

In an information filtering system, the system's representation of the information need (i.e., the compromised information need) is commonly referred to as a "profile." Because the profile fills the same role as what is commonly called a "query" in information retrieval and database systems, sometimes the term "query" is used instead of "profile" in information filtering as well. It would not be technically correct to call the profile a "user model" because a user model consists of both a representation of the user's interests and a method for interpreting that representation to make predictions. But that usage occasionally appears as well. We shall avoid confusion on this subject by using only the term "profile" when referring to the compromised information need in the context of information filtering. Table 2.2 summarizes the terminology that we have described in this section.

Our Term	Other Commonly Used Terms
Filtering	Routing, SDI, Current Awareness, Data Mining
Information Need	Interest, Visceral Information Need, Topic
Profile	Query, Compromised Information Need

Table 2.2: Information filtering terminology.

2.3 Historical Development

Luhn introduced the idea of a "Business Intelligence System" in 1958 [90]. In Luhn's concept, library workers would create profiles for individual users, and then those profiles would be used in an exact-match text selection system to produce lists of new documents for each user. Orders for specific documents would be recorded and used to automatically update the requester's profile. Foreshadowing later concerns about privacy, Luhn also observed that a set of profiles could be used to identify which users had expertise in specific areas.

Luhn's early work identifies every aspect of a modern information filtering system, although the microfilm and printer technology of the day resulted in significantly different implementation details. In describing the function of the selection module as "selective dissemination of new information" Luhn coined the term which described this field for nearly a quarter century.

A decade later, widespread interest in Selective Dissemination of Information (SDI) resulted in creation of the Special Interest Group on SDI (SIG-SDI) of the American Society for Information Science. Houseman's 1969 survey for that organization identified 60 operational systems, nine of which served over 1,000 users each [65]. These systems generally followed Luhn's model, although only four of the 60 implemented automatic profile updating, with the rest about evenly split between manual maintenance of the profiles by professional support staff or by the users themselves. Two factors had led organizations to make this investment in SDI: the availability of timely information in electronic form, and the affordability of sufficient computing capability to match those documents with user profiles. These are the same factors motivating information filtering today, although distribution of scientific abstracts on magnetic tape (the dom-

inant source of external information at the time) has been replaced by nearly instantaneous communications across large networks of interconnected computers.

Denning coined the term "information filtering" in the ACM President's Letter that appeared in the Communications of the ACM in March of 1982 [38]. Introducing the new ACM Transactions on Office Information Systems, Denning's objective was to broaden a discussion which had traditionally focused on generation of information to include reception of information as well. He described a need to filter information arriving by electronic mail in order to separate urgent messages from routine ones, and to restrict the display of routine messages in a way that matches the personal mental bandwidth of the user. Among the possible approaches he identified was a "content filter." The remaining six techniques (hierarchical organization of mailboxes, separate private mailboxes, special forms of delivery, importance numbers, threshold reception, and quality certification) all required the cooperation of the other users, and hence would better be studied from a more global perspective than the receiver's local scope of action represented by the information seeking model in figure 2.1. We shall have more to say on Denning's other approaches in section 2.5.3.

Over the subsequent decade, occasional papers on information filtering applications appeared in the literature. While electronic mail was the original domain about which Denning had written, subsequent papers have addressed newswire articles, Internet "News" articles,[3] and broader network resources [48,

---

[3]Internet "News" (more properly USENET News) is not a news source in the traditional sense, but rather a form of distributed electronic conference support system in which submissions (referred to as articles) are propagated to central repositories at participating institutions.

73, 115, 158]. The most influential paper of this period was published in the Communications of the ACM by Malone, *et al.* in 1987 [91]. There they introduced three paradigms for information selection, "cognitive," "economic," and "social," based on their work with a system they called the "Information Lens." Their definition of cognitive filtering, the approach actually implemented by the Information Lens, is equivalent to the "content filter" defined earlier by Denning, and this approach is now commonly referred to as "content-based" filtering. They also described an economic approach to information filtering, a generalization of Denning's "threshold reception" idea, that had implications beyond the scope of the information seeking system model in figure 2.1. We describe the economic issues related to information filtering briefly in section 2.5.3.

The most important contribution of Malone, *et al.* was to introduce an alternative approach which they called social (now also called "collaborative") filtering. In social filtering, the representation of a document is based on annotations to that document made by prior readers of the document. They speculated that by exchanging this sort of information, communities of shared interest could be automatically identified.[4] If practical, social filtering would provide a basis for selection of information items, regardless of whether their content could be represented in a way that was useful for selection. The balance between content-based and collaborative filtering is an important unresolved issue, and we will have much more to say on the relative merits of the two approaches in the sections that follow.

Large-scale government-sponsored research on information filtering also be-

---

[4]The principal difference between social filtering and Denning's more limited concept of "quality certification" is that annotations can be combined more flexibly in social filtering.

gan in this period. In 1989 the United States Defense Advanced Research Projects Agency (DARPA) sponsored the first of an ongoing series of Message Understanding Conferences (MUC) [84, 64]. The principal thrust of those conferences has been use of information extraction techniques to support the selection of messages. In 1990, DARPA launched the TIPSTER project to fund the research efforts of several of the MUC participants [58]. TIPSTER added an emphasis on the use of statistical techniques to preselect messages that could then be subjected to more sophisticated natural language processing. In TIPSTER, this preselection process is known as "document detection." In 1992 The National Institute of Standards and Technology (NIST) capitalized on this research by co-sponsoring (with DARPA) an annual Text REtrieval Conference (TREC) focused specifically on text filtering and retrieval [59].

So for the first decade after Denning identified networked information as an important application for filtering technology, information filtering was either addressed episodically or included as part of a broader research effort. Finally, in November of 1991, Bellcore and the ACM Special Interest Group on Office Information Systems (SIGOIS) jointly sponsored a workshop on "High Performance Information Filtering" that brought together a substantial quantity of research to establish a basis for the explosive growth the field has experienced in the past five years. Forty contributors examined the area from a wide variety of perspectives, including user modeling, information selection, application domains, hardware and software architectures, privacy, and case studies. A year later, in December of 1992, expanded versions of nine papers from that workshop appeared in a special issue of the Communications of the ACM [4, 5, 14, 49, 56, 87, 120, 140, 142].

2.4 Case Studies

The recent surge of interest in information filtering has actually contributed to the flood of information, since there is now more being published in the field than any single individual could hope to read. In part this results from the coincident adoption of the World Wide Web as a rapid means for the dissemination of academic work. Presently there are literally hundreds of documents about information filtering accessible through that medium.[5] In this section we describe the two dominant research paradigms, content-based and social filtering, and examine issues related to each. We have selected systems to discuss which highlight approaches that operate on behalf of the receiver (rather than the sender) and which illuminate the issues which are significant for adaptive multilingual text filtering.

2.4.1 Content-Based Filtering

With a research heritage extending back to Luhn's original work, the content-based filtering paradigm is the better developed of the two. In content-based filtering, each user is assumed to operate independently. As a result, document representations in content-based filtering systems can exploit only information that can be derived from document contents. Yan implemented a simple content-based text filtering system for Internet News articles in a system called SIFT [161].[6] Profiles for SIFT were constructed manually by specifying words to prefer or avoid, and had top be be updated manually if the user desired to

---

[5]Network-accessible resources on information filtering that are known to the authors are collected at http://www.ee.umd.edu/medlab/filter

[6]SIFT has been commercialized, and it is available at http://www.reference.com

change them. For each profile, twenty articles were made available each day in a ranked output format. Articles could be selected interactively using a World Wide Web browser. For users lacking interactive access, clippings (the first few lines of each article) could instead be sent by electronic mail. In that case selections were done without user interaction, so users were offered the option of defining a profile for an exact match text selection technique.

SIFT offered two facilities to assist users with profile construction. Users were initially offered an opportunity to apply candidate profiles against the present day's articles to determine whether appropriate sets of articles are accepted and rejected. If a substantial amount of information on that interest was present on Internet News that day, iterative refinement allowed the user to construct a profile which would move the appropriate articles to the top of the list. To facilitate maintenance of profiles over time, words which contributed to the position of each article in the ranked list were highlighted (a technique known as "Keyword in Context" or "KWIC") when using a World Wide Web browser to access the articles. By examining the context of words which occur with meanings that were unforeseen at the time the profile was constructed, users could select additional words which appeared in the same context to add to the list of words to be avoided.

Yan developed SIFT to study efficient algorithms for information filtering. In SIFT, large collections of profiles were compared to every article arriving on Internet News by a central server. Efficiencies were obtained by grouping profiles in ways that permit parts of the filtering process to be performed on groups of profiles rather than individually. SIFT made no distinction among the words appearing in an article, so words appearing in the newsgroup name

31

(i.e., the specific conference), the author's electronic mail address, the article title, the body of the article, included text, or the "signature" information that is routinely added to every document by some users were all equally likely to result in a high rank for a document.

Present multilingual text filtering systems also rely on the manual profile construction technique. The "Fast Data Finder," a product of Paracel, Inc., uses thousands of custom processing units that are optimized for operations such as term weighting, proximity constraints, and exact and fuzzy matching [94].[7] Each processing unit is programmed for a single task, and separate pipelines of processing units are formed for each profile. Simultaneous searching with multiple profiles is supported, so multilingual profiles can be implemented as a set of monolingual profiles, one for each language. Automated tools are provided to assist users with profile translation, so the effort expended to construct a profile in the first language can be leveraged to quickly produce profiles which will recognize the same concepts in other languages. But like SIFT, no provisions are made to automatically update Fast Data Finder profiles in response to the user's behavior.

Stevens developed a system called InfoScope which used automatic profile learning to minimize the complexity of exploiting information about the context in which words were used [143]. Like the electronic mail version of SIFT, InfoScope was also designed to filter Internet News using exact-match rules. InfoScope implemented adaptive filtering, however, suggesting rules based on observations of user behavior and offering them for approval (possibly with modifications) by

---

[7]Additional information on the Fast Data Finder is available from Paracel Inc., 80 South Lake Avenue, Suite 650, Pasadena, CA 91101-2616.

the user. These suggestions were based on simple observable actions such as the time spent reading a newsgroup or whether an individual message was saved for future reference. By avoiding the requirement for explicit user feedback about individual articles, InfoScope was designed to minimize the cognitive load of managing the information filtering system.

While SIFT treats Internet News as a monolithic collection of articles, Info-Scope was able to make fine-grained distinctions between newsgroups, subjects, and even individual authors. Implementation of such extensive deconstruction led Stevens to introduce a facility to reconstruct levels of abstraction in a way that was meaningful to the user. InfoScope implemented this abstraction at the newsgroup level, suggesting to combine related sets of newsgroups that were regularly examined by the user to form a single "virtual newsgroup." By defining filters for virtual newsgroups with possibly overlapping sources, users were thus provided with a powerful facility to reorganize the information space in accordance with their personal cognitive model of the interesting parts of the discussions they wished to observe.

InfoScope was not without its limitations, however. The experimental system Stevens developed was able to process only information in the header of each article (e.g., subject, author, or newsgroup), a restriction imposed by the limited personal computer processing power available in 1991. In addition, Stevens' goal of exploring the potential for synergy between user and machine for profile management led him to choose a rule-based exact match text selection technique. Since users are often able to verbalize the selection rules they apply, Stevens reasoned that users would have less difficulty visualizing the effect of changing rules than the effect of changing the types of profiles commonly found in ranked

output systems. InfoScope's key contributions, machine-assisted profile learning, the addition of user-controlled levels of abstraction, and implicit feedback, make it an excellent example of a complete content-based information filtering system intended for interactive use.

Because of their low cost, large volume, and ease of recognizing new information, Internet News and electronic mail have been popular domains for information filtering research. Unfortunately, these domains are poorly suited to formal experiments because reproducible results are difficult to obtain. For this reason, very little is known about the effectiveness of either SIFT or InfoScope. Stevens reported that eight of ten experienced Internet News readers preferred InfoScope to their prior software in an initial study, and that all five users in the second evaluation reported that fewer uninteresting articles were presented and more interesting articles were read in a second half of a 10 week evaluation than in the first. Because SIFT was developed to study efficiency rather than effectiveness issues, even less information is available about its effectiveness. Yan does report, however, that in early 1995 SIFT routinely processed over 13,000 profiles and was adding approximately 1,400 profiles each month [161]. Even though one user could create several profiles, this level of user acceptance makes a powerful statement about the utility of even the simple approach used by SIFT.

Learning more about the effectiveness of a text filtering technique requires that the technique be evaluated under controlled experimental conditions. And because the performance of text filtering techniques varies markedly when different information needs and document collections are used, comparison of results across systems is facilitated when those factors are held constant. The TREC evaluation has provided an unprecedented venue for exactly this type of perfor-

mance evaluation. Conducted annually since 1992, the most recent conference (TREC-4) attracted participation from 24 universities and 12 corporations [60].

NIST provides each participant with fifty topics and a large set (typically thousands) of training documents and relevance assessments for each topic.[8] Participants train their text filtering systems, using this data as if it represented explicit feedback on the utility of each training document to a user, and then must register their profiles with NIST before receiving the evaluation documents. The profiles are then used by the text filtering systems which generated them to rank order a previously unseen set of evaluation documents, and the top several thousand documents are submitted to NIST for evaluation.

In order to achieve reproducible results, it is necessary to make some very strong assumptions about the nature of the information filtering task. In TREC it is assumed that human judgements about whether an information need is satisfied by a document are binary valued (i.e., a document is relevant to an information need or it is not) and constant (i.e., it does not matter who makes that judgement or when they make it). Relevance, the fundamental concept on which this methodology is based, actually fails to satisfy both of those assumptions. Human relevance judgments exhibit significant variability across evaluators, and for the same evaluator across time. Furthermore, evaluators sometimes find it difficult to render a binary relevance judgment on a specific combination of a document and an information need. Nevertheless, performance measures based on a common set of relevance judgements provide a principled basis for comparing the relative performance of different text filtering techniques.

---

[8]Relevance assessments for the TREC "routing" (text filtering) training documents generally are derived from TREC text retrieval evaluations conducted in prior years.

The TREC filtering evaluation is based on effectiveness measures that are commonly used for text retrieval systems. The effectiveness of exact match text retrieval systems is typically characterized by three statistics: "precision," "recall," and "fallout." Precision is the fraction of the selected documents which are actually relevant to the user's information need, while recall is the fraction of the actual set of relevant documents that are correctly classified as relevant by the text filtering system. When used together, precision and recall measure selection effectiveness. Because both precision and recall are insensitive to the total size of the collection, fallout (the fraction of the non-relevant documents that are selected) is used to measure rejection effectiveness. Table 2.3 illustrates these relationships.

Selected as	Actually is	
	Relevant	Not Relevant
Relevant	Found	False Alarm
Not Relevant	Missed	Correctly Rejected

$$\text{Precision} \quad = \quad \frac{\text{Found}}{\text{Found} + \text{False Alarm}} \quad\quad\quad (2.1)$$

$$\text{Recall} \quad = \quad \frac{\text{Found}}{\text{Found} + \text{Miss}} \quad\quad\quad (2.2)$$

$$\text{Fallout} \quad = \quad \frac{\text{False Alarm}}{\text{False Alarm} + \text{Correctly Rejected}} \quad\quad\quad (2.3)$$

Table 2.3: Measures of text selection effectiveness.

In TREC, almost all of the filtering systems produce ranked output. Accordingly, precision and fallout at several values of recall are reported, and "average precision" (the area under the precision-recall curve) is reported for use when a

single measure of effectiveness is needed [129]. Average precision is computed by choosing successively larger sets of documents from the top of the ranked list that result in evenly spaced values of recall between zero and one. Precision is then computed for each set, and the mean of those values is reported as the average precision for an individual information need. The process is repeated for several information needs, and the mean of the values obtained is reported as the average precision for the system on that test collection. Clearly, larger values of average precision are better.

Only the selected documents must be scored to evaluate precision, but it would be impractical to evaluate recall and fallout by scoring every document in the TREC collection. The solution is to estimate recall and fallout by scoring a sample of the document collection. The approach chosen for TREC, known as "pooled relevance evaluation" is to evaluate every document chosen by any participating system and then assume that all unchosen documents are not relevant. Since documents are chosen using a wide variety of text filtering and retrieval techniques in TREC, it is felt that the pooled relevance methodology produces a fairly tight upper bound on recall and an extremely tight lower bound on fallout.

Although TREC investigates only the performance of the selection module, and that evaluation is necessarily based on a somewhat artificial set of assumptions, the resulting data provides a useful basis for choosing between alternative selection techniques. In the TREC-3 evaluation, for example, 25 text filtering systems were evaluated and average precision was observed to vary between 0.25 and 0.41.

2.4.2   Social Filtering

The Tapestry text filtering system, developed by Nichols, *et al.* at the Xerox Palo Alto Research Center (PARC), was the first to include social filtering [56, 150]. Designed to filter personal electronic mail, messages received from mailing lists, Internet News articles, and newswire stories, Tapestry allowed users to manually construct profiles based both on document content and on annotations made regarding those documents by other users. Those annotations were explicit binary judgements ("like it" or "hate it") that could optionally be made by each user on any message they read.

Like InfoScope, Tapestry profiles consisted of rules that specified the conditions under which a document should be selected. One important difference was that Tapestry allowed users to associate a score with each rule. Tapestry then generated ranked output by comparing the scores assigned by multiple rules. Tapestry implemented this sophisticated processing efficiently by dividing the filtering process into two stages using a client-server model. In the first stage, a central server with access to all of the documents applies a set of simple rules, similar to those used by SIFT, to determine whether each document may be of interest to each user. The more sophisticated rules in each profile are then executed in each users' workstation (the client) to develop the ranked list.

Experience with several small scale trials of social filtering suggests that a critical mass of users with overlapping interests is needed for social filtering to be effective. Tapestry was restricted to a single site because both the content and the software were subject to proprietary restrictions, so only limited anecdotal evidence of the social filtering aspects of Tapestry's performance are available. From this experience and others (c.f., [15, 63, 137]) it appears that social filtering

systems must assemble a fairly large critical mass of users before it would be possible to demonstrate their effectiveness. The ongoing GroupLens project of Miller, *et al.* at the University of Minnesota is presently the most ambitious attempt to reach such a critical mass using an information filtering system that is designed to manage a dynamic information source [123].

GroupLens is designed to filter Internet News, a freely redistributable text source. Like Tapestry, GroupLens is built on a client-server model. GroupLens uses two types of servers, content servers (which are simply standard Internet News servers) and annotation servers (which have been developed for the project). The design permits both the content and annotation servers to be replicated so that each server can efficiently service a limited user population. Modified versions of some popular (and freely redistributable) Internet News client software are made available in order to encourage the development of a large user population, and implementers of other client software are permitted to incorporate the GroupLens protocol in their products.[9]

GroupLens annotations are explicit judgements on a five-valued integer scale. Unlike Tapestry, however, the annotations need not be assigned an *a priori* interpretation. Users may register annotations with their annotation server using whatever semantics for the five values they wish. The annotation servers collect annotations from their user population, use correlation information to predict their user evaluations of unseen articles, and provide those predictions to client programs on request. The initial GroupLens trial began in 1996 using a limited number of newsgroups and a single annotation server. Results are not yet avail-

---

[9]The GroupLens protocol and GroupLens client software can be obtained from http://www.cs.umn.edu/Research/GroupLens

able, but the project's important contributions, distributed annotation servers, profile learning for social filtering, and a design which encourages development of a large user base, provide an excellent prototype for future work on social filtering.

One limitation of the existing experimental work on social filtering is user motivation. In GroupLens, users annotate documents in order to improve the performance of their filter's ability to learn from other clients who have annotated the same documents. This creates a bit of a "chicken and the egg" problem, though, since there is no incentive for the first user to annotate anything. If content-based and social filtering are integrated in the same system, however, then a synergy between the two techniques can develop. Tapestry demonstrated one way in which the two approaches can be combined when manually constructed profiles are used. The URN system, developed by Brewer at the University of Hawaii, illustrated a more automatic method by which such synergy can be achieved.

URN was an Internet News filtering system in which users could provide two types of information to support profile learning [15]. The first was by making explicit binary judgements about the utility of the document. Those judgements were then used as a basis for a typical content-based ranked output system. What makes URN unique is that users can also collaboratively improve the system's initial representation of the document by adding or deleting words which they feel represent (or, for deletions, misrepresent) the content of the document. In URN these changes are propagated to all other users, allowing the user community to collaboratively define the structure of the information space. Since user-specified words are given preference by URN when developing representations

for new documents, users have an incentive to improve the set of words which describe existing documents.

In URN each user maintains a separate content-based user model, while the annotation server effectively maintains a single collaboratively-developed model of the document space. This approach lacks the sophistication of the separate user models based on shared annotations found in GroupLens, but URN's integration of content-based and social filtering techniques illustrates one way in which these two paradigms can be combined.

2.5 Text Filtering Technology

As we have described, the essence of text filtering practice is not the techniques themselves, but rather the way in which those techniques are integrated to support a specific text filtering process. But just as adaptive multilingual text filtering systems are constructed using techniques inspired by two other fields (adaptive text filtering and multilingual text retrieval), text filtering itself draws from a number of different disciplines. In this section we identify the sources of techniques which have been synthesized to produce effective and efficient text filtering systems. Since these techniques are drawn from several fields, our presentation considers each field in turn. Because we seek here to describe present practice, we defer to Chapter 4 a detailed discussion of the multilingual text retrieval techniques which have informed our extension of these techniques to support adaptive multilingual text filtering.

2.5.1 Information Retrieval

As Belkin and Croft observed, content-based text selection techniques have been extensively evaluated in the context of information retrieval [5]. Every approach to text selection has four basic components:

- Some technique for representing the documents

- Some technique for representing the information need (i.e., profile construction)

- Some way of comparing the profiles with the document representations

- Some way of using the results of that comparison

The objective is to automate the process of examining documents by computing comparisons between the representation of the information need (the profile) and the representations of the documents. This automated process is successful when it produces results similar to those produced by human comparison of the the documents themselves with the actual information need. The fourth component, using the results of the comparison, is actually the role of the display module in figure 2.1. We include it here to emphasize the close coupling between selection and display.

In each of the text filtering systems we describe in this chapter, the selection module assigns one or more values to each document, and the display module then uses those values to organize the display. Figure 2.3 illustrates the representation and comparison process implemented by those systems. The domain of the profile acquisition function p is I, the collection of possible information needs and its range is R, the unified space of profile and document representations.

The domain of the document representation function d is D, the collection of documents, and its range is also R. The domain of the comparison function c is $R \times R$ and its range is $[0,1]^n$, the set of n-tuples of real numbers between zero and one. In an ideal text filtering system,

$$c(p(\text{info need}), d(\text{doc})) = j(\text{info need}, \text{doc}), \forall \text{info need} \in I, \forall \text{doc} \in D, \quad (2.4)$$

where $j : I \times D \mapsto [0,1]^n$ represents the user's judgement of some relationships between an interest and a document, measured on n ordinal scales (e.g., topical similarity or degree of constraint satisfaction).
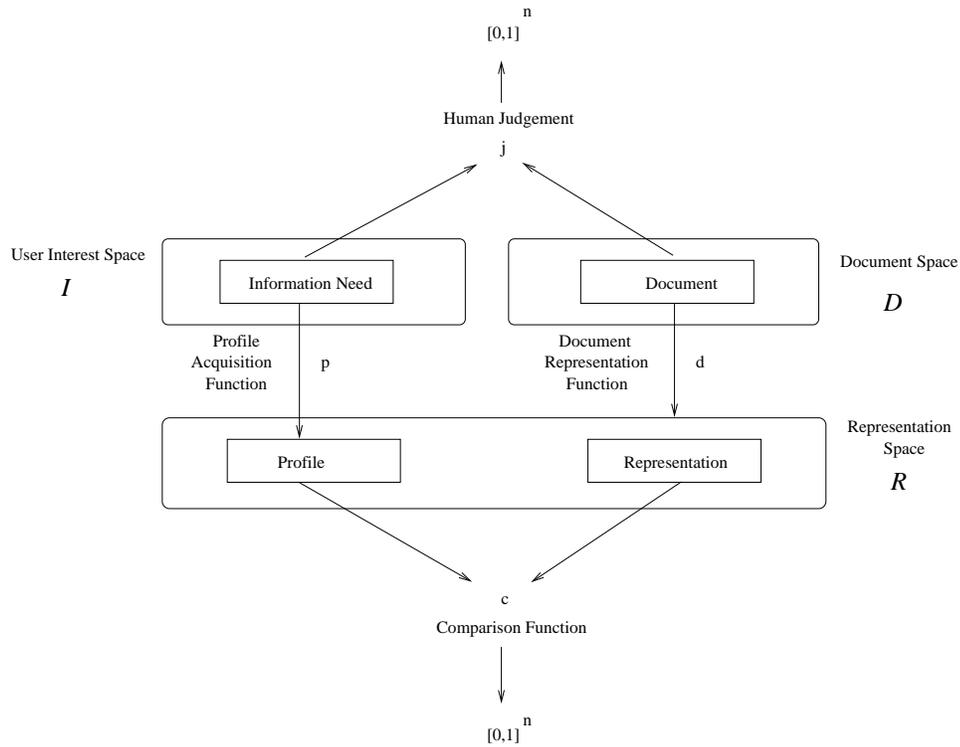
Figure 2.3: Text filtering system model.

As we saw in section 2.4, the representation can exploit information derived from the content of the document, annotations made by others, or some combination of the two. Although syntactic and semantic analysis of documents is

possible, content-based text filtering systems typically use representations based on the frequency with which terms occur in each document.[10] One reason for this choice is that it lends itself to efficient implementation. But a more compelling reason is that because no domain-specific information is needed to form the representation, a demonstration of acceptable performance in one application is easily translated into similar performance in another.

Although content-based text filtering systems typically start with this term-frequency representation, they generally apply some type of transformation to that representation before invoking the comparison function c in figure 2.3 The nature of the transformation depends strongly on which characteristics of that representation the comparison function c is designed to exploit, however. For this reason, we describe the transformations together with their associated comparison functions in the following paragraphs.

For an exact match text filtering system the range of the comparison function c is restricted to be either zero or one, and it is interpreted as a binary judgement about whether a document satisfies the profile. In this case, a step function that detects term presence is applied to the term-frequency representation when that representation is constructed so that the resulting boolean vector can be easily compared to the boolean expression specified by the profile. Exact match text filtering systems typically provide an unranked set of documents which will (hopefully) satisfy the information need. The exact match approach is well suited to autonomous systems which must take actions (such as storage decisions)

---

[10]We use "terms" rather than "words" because the "terms" which are considered may be parts of words (e.g., overlapping three letter subsequences known as trigrams), single words, or combinations of words (e.g., idiomatic phrases). Common "stopwords" that have little use in subsequent processing are typically eliminated during term selection.

without user interaction.

Two common approaches to ranked output generation are the vector space method and the probabilistic method, although variations abound. In the vector space method the range of c is [0,1], and the value is interpreted as the degree to which the content of two documents is similar. Both the profile and the documents are represented as vectors in a vector space, and a comparison technique based on the assumption that documents whose representations are similar to the profile will be likely to satisfy the associated information need is used. The angle between two vectors has been found to be a useful measure of content similarity, so the the square of the cosine of that angle (easily computed as the normalized inner product of the two vectors) is used to rank order the documents.

$$\cos(v_1, v_2) = \sqrt{\frac{v_1^{\mathrm{T}} v_2}{\sqrt{v_1^{\mathrm{T}} v_1}\sqrt{v_2^{\mathrm{T}} v_2}}} \tag{2.5}$$

The vector space method's effectiveness can be improved substantially by transforming the raw term-frequency vector in ways which amplify the influence of words which occur often in a document but relatively rarely in the whole collection [51]. One common scheme, known as "term-frequency—inverse document frequency" weighting, assigns term i in document k a value computed as:

$$tfidf_{ik} = \text{occurrences of term } i \text{ in doc } k * \ln(\frac{\text{number of docs}}{\text{number of docs with term } i}) \tag{2.6}$$

In a text filtering system, advance knowledge of the inverse document frequency portion of that equation is clearly not possible. Estimates of that information based on sampling earlier documents can, however, produce useful inverse document frequency values for domains in which term usage patterns are relatively stable.

Rather than estimate similarity, the probabilistic method seeks to estimate the probability that a document satisfies the information need represented by the profile. The probabilistic method is thus a generalization of the exact match technique in which we seek to rank order documents by the probability that they satisfy the information need rather than by making a sharp decision. To develop this probability, term frequency information (weighted to emphasize within document frequency and to deemphasize across-document frequency) is treated as an observation, and the distribution of the binary event "document matches profile" conditioned by that observation is computed. Bayesian inference networks have proven to be a useful technique for computing this conditional probability [152]. Since it is possible to construct a Bayesian inference net which computes the cosine of the angle between two vectors, the vector space method can be interpreted as a special case of the probabilistic method [153].

Since the comparison function can produce a multiple-valued result, the display module can be designed to exploit the results of both exact match and ranked output techniques. For example, an electronic mail system could reject documents sent by specific users and then rank the remaining documents in order of decreasing content similarity to a prototype document provided by the user. The profile is what Korfhage has called a "reference point" and together the profile and the comparison technique in a ranked output text filtering system can be thought of as specifying a "point of view" in the document space [79]. Multiple rank orderings can be combined to produce richer displays that combine multiple points of view, a research area often referred to as "document visualization" or "visual information retrieval interfaces."

Although only the vector space method actually uses vector operations such as the inner product, all three of these approaches exploit "feature vectors" in which the features are based on the frequency with which terms appear within documents and across the collection. The annotations provided by social filtering techniques are an additional source of features that can be exploited by a comparison function. Because annotations can be used even when useful content-based features are difficult to construct, information retrieval systems designed for information that is not in text form have explored matching techniques for feature vectors composed of annotations.

One such application which appears to have reached the critical mass necessary for effective use of annotations is a home video recommendation service developed by Hill, *et al.* at Bellcore in which users' tastes in movies were matched using techniques similar to those implemented in GroupLens [63]. Populated with a large and relatively stable set of movie titles, stable interests could be matched against that database for some time before exhausting the set of movies that might be of interest to a user. This is an interesting case in which the unlabeled corner of the graph in figure 2.1 is worth exploring.

Hill's system allowed users to provide numeric evaluations (on a scale of one to ten) for movies they had already seen, and then matched those ratings with evaluations of the same movies that had previously been provided by other users. Movies were sorted by category (e.g., drama or comedy), and within a category correlation coefficients between the feature vectors were computed. A set of users with the largest correlations was then selected and regression was performed based on evaluations from those users to predict scores for unseen movies in each category. In this case the profile was the set of annotations

provided by the user, the "document" features were the annotations provided by others, and the comparison function was a two-step process of feature selection followed by regression.

In addition to showing how annotations can be viewed as features, this example illustrates an important limitation of the information retrieval techniques we have described. In information filtering applications, profiles based on multiple documents (such as the multi-movie evaluation within a category used in Hill's system) are common. But information retrieval research has explored only relatively simple ways of combining this information to form profiles. Relevance feedback, an information retrieval technique in which feature vectors are formed from the content of multiple documents, has shown good results. But the "one query at a time" model which underlies much information retrieval research precludes consideration of techniques such as the regression used by Hill, *et al.*

2.5.2 User Modeling

Machine learning, the study of algorithms that improve their performance with experience, offers a source of techniques that are designed to exploit multiple training instances to improve selection effectiveness [82]. Machine learning is one component of "user modeling," a discipline which is concerned with both how information about users can be acquired and used by automated systems.[11] The models we consider in this chapter are what Rich has called "individual

---

[11]As Karlgren, *et al.* have observed, it is also important to construct systems whose operation conforms with the user's mental model of the information filtering process [76]. The user models we refer to in this chapter, however, are models constructed by the system which describe some aspect of the user.

user, long-term user models" [124].

Sources of Information About the User

Before describing how machine learning techniques have been applied to text filtering it is useful to consider more carefully how information about the user can be acquired. Rich defined a distinction between "explicit" models which are "constructed explicitly by the user" and "implicit" models which are "abstracted by the system on the basis of the user's behavior" [124]. Both implicit and explicit user models are found in text filtering systems (SIFT, for example, uses an explicit model). The machine learning techniques we describe in section 2.5.2 can be used to create what Rich called implicit models.

In order to construct an implicit user model we must be able to observe both the user's behavior and the salient features of the environment in which that behavior is exhibited. In the case of text filtering, the salient elements of the environment are the documents which have been examined by the user. Section 2.5.1 described how information about those documents can be acquired, either from contents or from annotations made by others.

In section 2.4 we presented several examples of how representations of previously seen documents can be combined with evidence of the user's interest in those documents to predict interest in future documents. With the exception of InfoScope, every system we have described requires the user to explicitly evaluate documents, a technique we refer to as "explicit feedback."[12] Explicit feedback

---

[12]There is some potential for confusion here because we are describing the use of explicit feedback to construct what Rich has called an implicit user model. In order to minimize confusion, we avoid using the terms "implicit" and "explicit" in isolation.

has the advantage of simplicity. Furthermore, in experimental systems explicit feedback has the added advantage of minimizing one potential source of experimental error, inference of the user's true reaction. But in practical applications explicit feedback has two serious drawbacks. The first is that a requirement to provide explicit feedback increases the cognitive load on the user. This added effort works against one of the principal benefits of a text filtering system, the reduced cognitive load that results from an information space more closely aligned with the user's perspective. This problem is compounded by the observation that numeric scales may not be well suited to describing the reactions humans have to documents. For example, is a document which address the information need well but contains little expository text better or worse than a document that is easily understood but less complete? These difficulties motivate the study of implicit feedback mechanisms.

In Stevens' InfoScope system, three sources of implicit evidence were observed about the user's interest in each message: whether the message was read or ignored, whether it was saved or deleted, and whether it was replied to or not. Because the users decision to read or ignore the message was necessarily based on a summary of the same message header information that InfoScope used to construct feature vectors, it would be reasonable to assume that the "read or ignore" decision would be nearly as useful as explicit feedback. InfoScope did, however, allow explicit feedback as well.

Morita and Shinoda also investigated implicit feedback for filtering Internet News articles, using both save and reply evidence but substituting reading duration for InfoScope's "read or ignore" evidence [96]. In a six week study of eight users, they found a strong positive correlation between reading time and explicit

feedback provided by the user on a four-level scale. Furthermore, they discovered that interpreting as "interesting" articles which the reader spent more than 20 seconds reading produced better recall and precision in a text filtering experiment than using documents explicitly rated by the user as interesting. This surprising result reinforces our observation that users sometimes have difficulty expressing their interest explicitly on a single numeric scale.

Since the experimental subjects were asked to read articles without interruption, it is not clear whether such useful relationships can be found in environments where reading behavior is more episodic. But Morita and Shinoda's results, coupled with the anecdotal evidence reported by Stevens, suggest that implicit feedback may be a practical source of features to which machine learning algorithms can be applied. Both implicit and explicit feedback produce features that are associated with documents. But unlike the feature vectors which describe the document's contents, feature vectors based on implicit or explicit feedback describe the user's reaction to the document.

Machine Learning

Complete feature vectors describing both the document and the user's reaction to it can be constructed for documents which have been read by adjoining the features that represent the document (e.g., term frequency values) with the vector that represents the user's reaction to it (e.g., explicit feedback). For new documents, only those features that represent the document will be known, and it would clearly be useful to be able to estimate the missing information (the user's anticipated reaction to the document). In the field known as "machine learning" this is known as the "supervised learning" problem.

In the canonical supervised learning problem, the machine is presented with a sequence of feature vectors (training instances), and then it is required to predict one or more missing elements in another set of feature vectors.[13] Predicting these missing values is an induction process, so induction forms the basis for machine learning. No induction technique can be justified without reference to domain knowledge, however. Because it would be possible to explain any set of observations after the fact, in the absence of some bias in the induction technique, any values could reasonably be predicted.[14] Langley identifies three ways in which this necessary bias can be introduced in a machine learning system: in the representation, in the search technique, and as explicit domain knowledge [82]. The vector space method, in which profiles are represented as a single vector and documents are ranked based on the angular similarity of their representation with that vector, combines both representation bias and search bias. InfoScope's learning heuristics (e.g., suggest filters for newsgroups that are read in at least 2 of the most recent 6 sessions) is an example of domain knowledge bias.

Supervised learning is particularly well suited to exact match filtering systems which use explicit binary feedback, because in that case the training data contains exactly the same information (whether or not to select a document) that must be estimated for newly arrived documents. This is a special case of the "classification" problem, in which we wish to sort newly arrived documents into two or more categories (in this case, retained and rejected). Supervised

---

[13]What we describe here is actually a restricted case of the supervised learning problem that is specialized to vector representations.

[14]One possible "after the fact explanation" would simply be that the formerly unknown parameters are random variables with some (still unknown) distribution that included the observed values.

learning can also be applied in ranked output filtering systems that use explicit feedback, assigning as a score for each document the system's estimate of the score that the user would assign. When implicit feedback is used, the ranking can be based on the predicted value of some observed parameter (e.g., reading duration). Alternatively, a manually constructed user model can be used to combine several observed parameters to produce an estimate of utility and then that estimate can be used to augment the training data.

Six classic machine learning approaches have been applied to text filtering: rule induction, instance based learning, statistical classification, regression, neural networks, and genetic algorithms. Stevens' work on InfoScope is an example of rule induction. InfoScope's filter suggestions were implemented as a decision list of parameters (newsgroup, field and word) which, if present in an article, would result in either selection or rejection of that article. These rules (e.g., select if newsgroup is rec.sewing and "bobbin" appears in the subject field) are learned using heuristics which can be modified by the user.

Foltz applied an instance based learning technique to selection of Internet News articles [48]. Representations of about 100 articles from a training collection which the user designated as interesting were retained, and then new articles were ranked by the cosine between their representation and the nearest retained representation. In other words, articles were ranked most highly if they were the most similar (using the cosine measure) to some positive example. In a small (four user) study, Foltz found that this technique produced an average precision of 0.55 (43% above that achieved by random selection), and that a further improvement to 0.61 (11%) could be achieved using a dimensionality reduction technique known as Latent Semantic Indexing (LSI).

This dimensionality reduction is an example of "feature selection." Feature selection can be an important issue when applying machine learning techniques to vector representations. Langley has observed that "many algorithms scale poorly to domains with large numbers of irrelevant features," [82] and it is not uncommon to have thousands of terms in the vocabulary of a text filtering system. Schütze, *et al.* at Xerox PARC applied two rank reduction techniques, one using the best 200 terms found with a χ^2 measure of dependence between terms and relevant documents, and the other using a variation of the LSI dimension-reduction technique used by Foltz [132]. For each of these feature selection techniques they evaluated four machine learning techniques, linear discriminant analysis (a statistical decision theory technique), logistic regression, a two-layer (linear) neural network, and three-layer (nonlinear) neural network using training and evaluation collections from TREC.

Schütze, *et al.* found that using only the LSI feature vectors provided the best filtering effectiveness with linear discriminant analysis and with logistic regression, and that their implementation of linear discriminant analysis was the better of the two techniques. They also found that both the linear and nonlinear networks were able to equal the effectiveness of linear discriminant analysis on the LSI feature vectors, but that both types of networks performed slightly (but not statistically significantly) better when presented with both sets of selected features simultaneously. Finally, they found that a nonlinear neural network resulted in no improvement over their simpler linear network. We explore the use of LSI feature vectors in detail in Chapter 3 because they have a natural extension to multilingual applications.

Exploring another machine learning technique, Sheth implemented a genetic

algorithm to filter Internet News in a system called "Newt" [137]. A genetic algorithm uses algorithmic analogues to the genetic crossover and mutation operations to generate candidate profiles that inherit useful features from their ancestors, and uses competition to identify and retain the best ones. Candidate profiles in Newt were vectors of term weights.[15] Relevance Feedback based on explicit binary evaluations of articles was used to improve candidate profiles, moving them closer in the vector space to the representation of desirable articles and further from the representation of undesirable ones. In machine learning this approach is referred to as "hill climbing." The crossover operator was periodically applied to combine segments of two candidate profiles which were among those that had produced the highest ranks (using a cosine similarity measure) for articles that the user later identified as desirable. A mutation operator was sometimes applied to the newsgroup name to explore whether existing candidate profiles would perform well on newsgroups with similar names. All of the candidate profiles contributed to the ranking of the documents shown to the user, although those which consistently performed well contributed more strongly to the ranking. Hence, the profile itself was determined by the population of candidate profiles, rather than by any individual candidate.

Sheth evaluated Newt using a technique referred to in machine learning as a "synthetic user." By generating (rather than assessing) user preferences, the synthetic user technique allows specific aspects of a machine learning algorithm's performance (e.g., learning rate) to be assessed. Sheth created synthetic users whose interests were deemed to be satisfied whenever at least one word from a list

---

[15]In Newt, terms were segregated by the field of the article in which they occurred, so "talk" in the subject field could be assigned a different weight than "talk" in the body of a message.

associated with that simulated user appeared in an article. Using this technique
he found that although individual candidate profiles were able to learn to sat-
isfy a simulated user quickly, when the simulated user's interest shifted abruptly
(simulated by changing the list of words associated with the simulated user)
individual candidate profiles were slower to adapt. When evaluating complete
profiles made up of populations of individual candidates, Sheth demonstrated
the ability to control the adaptation rate by adjusting parameters of the genetic
algorithm. Experiments based on simulated users provide less insight into over-
all system-level performance than do experiments that are grounded in human
relevance judgements, so we do not use simulated users in the experiment we
report in Chapters 3 and 5. But because the technique is both economical and
reproducible, it can be useful when answers to specific questions about learning
performance are sought.

2.5.3 Other Fields

This completes our description of the two major sources of technology for text
filtering systems: information retrieval and user modeling. Humans pursue the
information filtering process in a social context, though, and the machines that
they use must operate in some physical context. In this section we briefly identify
the issues raised by the interaction between the information filtering process and
these larger contexts.

Networked Computing Infrastructure

The physical context for the information filtering process is the existing net-
worked computing infrastructure. The relevant portion of the physical context

may consist of, for example, isolated workstations monitoring a common news-feed, a workgroup computing environment supported by an intranet, or the entire Internet. With a few notable exceptions (SIFT and Tapestry), in our descriptions we have placed more emphasis on effectiveness than efficiency when describing design features and performance evaluations. This is not surprising, since most experimental work on text filtering has sought to demonstrate effectiveness and a small user population suffices for that purpose. Even the TREC evaluation, which requires filtering hundreds of thousands of pages of text, specifies only 50 topics each year.

Once adequate effectiveness has been demonstrated for small user populations, the task of engineering efficient implementations for widespread use of such systems remains. One alternative is to simply replicate the filtering system and then provide all of the content to each filtering system. Tapestry implemented a more sophisticated approach, demonstrating that an appropriate division of effort between server-side and client-side computing can improve overall efficiency.

In general, the goal of distributed computation is to optimize the tradeoff between distributing the workload and minimizing communication requirements. Yan studied this issue rigorously in conjunction his with work on SIFT, developing optimal assignments of computational tasks among a group of cooperating servers [160]. The GroupLens project has chosen an alternative approach that exploits an existing infrastructure for document distribution. By augmenting this infrastructure with distributed annotation servers, GroupLens expects to achieve acceptable efficiency in a manner compatible with the existing physical and social structure for Internet News. Thus, one of the key issues to be ad-

dressed as the number of users scales up is which constraints to accept and which to change.

Computer Supported Cooperative Work

The same type of tension between constrained and unconstrained system design occurs at many levels. Adopting an even broader perspective, it is apparent that users operate within a social system, and that system imposes social constraints on what is possible. Organizational aspects of networked communications are studied the field of Computer Supported Cooperative Work (CSCW), so text filtering is an issue for which the CSCW perspective can be informative.

Consider, for example, Denning's suggestion that users set up separate mailboxes for specific purposes and that senders direct electronic mail to the appropriate mailbox. In order to be effective, this approach would require that the user address messages correctly, that receivers organize their mailboxes in a useful manner, and that all of the software systems between the sender and the receiver support this addressing scheme. Standards that are developed by consensus or through competitive market mechanisms often address such issues, and there are numerous examples of the practicality of such schemes (e.g., Lotus Notes and Internet News). Because many of the constraints on such efforts are social rather than technical, the breadth offered by the CSCW perspective is essential to the success of such endeavors.

Once such social conventions are created to add the necessary structure to the documents, text filtering techniques provide a way to exploit that information. For example, the current interest in assigning "ratings" to World Wide Web pages to facilitate parental control of the information available to their children

presumes the availability of technology to exploit that information. The design of a system for creating, distributing, and using these ratings is an issue best studied from the perspective of CSCW because a common task motivates multiple participants. Ratings are, however, simply one type of annotation. So an understanding of how annotations are used in information filtering systems can provide useful insight into how those annotations could be integrated with other sources of information about the contents of a document.

Market Formation

For applications which lack a shared objective, economic theory provides a more useful perspective than CSCW. In a market economy, "cost" or "price" (the value discovered by a market) serves as a basis for allocating scarce resources. In the emerging information-based economy, both information itself and the tools which manage that information have economic value. This will result in the development of a market for not merely information and tools, but also for metainformation such as the annotations on which social filtering is based. The CSCW perspective will certainly be helpful when designing common standards for the exchange of price information and monetary instruments because all participants in a market benefit from such social structures. But when participants do not share common goals with respect to the use they make of the information they obtain, market dynamics provide a more effective way of allocating scarce information resources such as intellectual property and expert annotations.

The vast majority of experimental work on text filtering has exploited freely available information such as Internet News and messages sent to electronic mailing lists, so little reference to the cost of intellectual property can be found in

that literature. On the other hand, users of commercial text filtering systems have developed profile construction techniques which which recognize differing costs for different aspects of access to intellectual property (e.g., selective purchase of limited redistribution rights) [39]. Commercial text filtering systems typically require explicit profiles, however, and we are not aware of any research on implicit user models for text filtering which exploit cost information. Like the ratings we described in section 2.5.3, prices are a type of annotation, and hence they can be exploited by a social filtering system. The difference between prices and other annotations on which social filtering can be based is that there may be a firmer *a priori* basis for using cost information than for using other types of annotations, and that fact may prove useful when designing user models for text filtering.

In addition to these technical considerations, market formation also raises broad social issues. The creation of markets for information, for annotations, and even for the filtering systems themselves restricts information access to users for whom the value of the information justifies the cost of obtaining it. Such unrestrained market operation is rarely allowed, however. Governments and other social structures are often charged with regulation of economic activity in order to limit the effect of inequities that can result from market economics. The establishment of public libraries, the imposition of disclosure requirements for securities transactions, and the regulations which subsidize universal access to the telephone network with revenue generated from other sources provide instructive examples of how market forces can be adjusted to accomplish social goals. If information truly has value then such issues of equity will undoubtedly arise in information filtering as well.

Privacy

Privacy becomes an issue when a system collects information about its user, so important social issues arise on an individual scale as well. In commercial applications, for example, it may be desirable to restrict access to profile information in order to protect a competitive advantage. And users with personal applications may demand that their profile remain private simply on moral grounds.

For content-based filtering systems, the privacy issue has two aspects: preventing unauthorized access to the profile and preventing reconstruction of useful information about the profile. The first issue is a straightforward security problem for which a variety of techniques such as password protection and encryption may be appropriate depending on the nature of the anticipated threat. But preventing reconstruction of useful information about the profile is a much more subtle problem. In Tapestry, for example, it would be possible to infer a good deal of information about the profile registered at the server by simply noting which documents were forwarded. An unauthorized observer who can detect which documents are being forwarded to specific users could conceivably build a second text filtering system (e.g., a social filter with an implicit user model) and then train it using the observed document forwarding decisions. Preventing such an attack would require that unauthorized observers be denied access to information about the sources and destinations of individual messages. In the computer security field, this is known as the "traffic analysis problem," and cryptographic techniques which address it have been devised (c.f., [23, 25]).

In the case of collaborative filtering, the situation is further complicated by the imperative to share document annotations. A simple approach (which is used by GroupLens) is to allow each user to adopt a pseudonym. While use of

61

pseudonyms makes it more difficult to associate annotations with users, traffic analysis can still be used to determine which users would read a document. Unfortunately, information about who is reading specific documents is exactly what other authorized users must know to perform social filtering. Furthermore, Hill has observed that users choosing which information to examine may find it useful to know the identity (not merely the pseudonym) of the users who made the annotations [63]. While encrypted transmission of annotations to other authorized users is a possibility in such cases, significantly limiting the user group in that way may prevent a social filtering system from reaching the necessary critical mass. This tension between a desire for privacy and the benefit of free exchange of information may ultimately limit the applications to which social filtering can be applied.

The level of protection which must be afforded to privacy varies widely across applications. By common agreement, many details of our private lives (e.g., birth, marriage and death) are a matter of public record. On the other hand, in some states of the United States it is a crime to divulge the borrowing history of a library patron without a court order. One can even envision applications in which a user might prefer not to know information represented in their own profile. Where these lines should be drawn is a matter of judgement that must ultimately be resolved by those who control the information resources that are being used.

2.6 Observations on the State of the Art

With this background we can now identify issues which will be important for further progress in the development of text filtering systems. In this section we first briefly address some large-scale issues which offer useful alternative perspectives from which the information filtering problem generally, and the text filtering problem in particular can be viewed. We then discuss in detail the relationship between the content-based and collaborative approaches, an issue which will be particularly important for the development of adaptive multilingual text filtering systems.

Early information filtering systems (then known as SDI) were developed to exploit the availability information in electronic form to manage the process of disseminating scientific information. When the printed page was the dominant information paradigm for text transmission, high production costs led to the development of extensive social structures (e.g., the peer review process) for selecting information worthy of publication. As long as this situation persisted, the dissemination process managed admirably, and SDI improved its performance. With the introduction of personal computing and ubiquitous networking, each participant is now able to also be both a consumer and a producer of information. The drastic reduction in publishing costs has greatly increased the importance of filtering the resulting flood of information, but the resulting variability in quality has also made that filtering task more difficult. Automatic techniques are needed to make this wealth of information accessible, since information that cannot be found is no better than information which does not exist.

Rather than simply removing unwanted information, information filtering actually gives consumers the ability to reorganize the information space [143].

For economic reasons, information spaces have traditionally been organized by producers and, in some cases, reorganized by intermediaries. In book publishing, for example, authors and publishers work together to assign titles to books and to announce their availability. Intermediaries such as libraries, book clubs and book stores obtain those announcements, select items which are likely to be of interest to their customers, and organize information about their selections in ways that serve the needs of those customers. Because such intermediaries typically serve substantial numbers of customers, economic factors usually limit them to providing a few (sometimes only one) perspectives on the information space.

Information filtering is essentially a personal intermediation service. Like a library, a text filtering system can collect information from multiple sources and produce an organization that is useful to the patrons. But by automating the process of organizing the information space it becomes economically feasible to personalize this organization. Of course, automating this intermediation process eliminates the value that could be added by human intermediaries who can apply their judgement to improve the organization of the information space.

Social filtering offers a way of integrating human and automated intermediation. Human intermediaries have traditionally organized the information space through selection and annotation. Selection, however, is simply a special type of annotation (i.e., a document is marked as "selected by the intermediary"). As with price annotations, the user may find it useful to assign expert annotations an *a priori* degree of confidence because they come from a source with well understood characteristics. Tapestry's profile specification language provides an example of how such functionality could be incorporated.

Social filtering is also inherently well suited to managing multilingual information because the representation is based on annotations (which need not be expressed using language at all) rather than content. But social filtering alone is unlikely to provide a complete solution to users' information filtering needs. Expert annotations require effort and have economic value, so the marketplace will undoubtedly assign them a price. With continued reductions in the cost of computing and communications resources, content-based filtering will offer a competitive source of information on which to base selections. Furthermore, because humans and machines base their evaluations on different features, systems which incorporate both social and content-based filtering will likely be more effective than those which use either technique in isolation. In this light, the work of Schütze, *et al.* suggests that machine learning techniques which effectively exploit multiple sources of evidence can be found [132].

Content-based and social filtering will almost certainly prove to be complementary in other, less easily measured ways as well. A perfect content-based technique would never find anything novel, limiting the range of applications for which it would be useful. Social filtering techniques excel at identifying novelty (because they are guided by humans), but only when the humans who guide them are not overloaded with information. Content-based systems can help to reduce this volume of information to manageable levels. Thus, both content-based and collaborative filtering contribute to the other's effectiveness, allowing an integrated system to achieve both reliability and serendipity. The focus of our work has been content-based filtering because that is where multilingual content can be directly exploited. But until these content-based approached are integrated with the collaborative approaches, many information information

filtering applications will likely fail to achieve their full potential.

One reason that this integration is not presently practical is that social filtering itself has yet to realize its potential. The difficulty of achieving a critical mass of participants makes social filtering experiments expensive. One clear disincentive in present experiments is the additional cognitive load imposed on the user by the requirement to provide explicit feedback. We are not aware of any research in which implicit feedback has been applied to social filtering, but there is some evidence that such an approach could be successful. Hill, *et al.* have reported that readers find it useful to know which portions of a document receive the most attention from other readers. In an analogy to the tendency of well-used paper documents to acquire characteristics which convey similar information, they call this concept "read wear" [62]. Coarser measurements such as Morita and Shinoda's reading time metric, or the save and reply decisions explored by Stevens, may also prove to be useful bases for social filtering in some applications. If useful annotations can be acquired without requiring explicit feedback, lesser inducements (such as the improvement that could result from application of a simple content-based filtering technique) may be sufficient to assemble the critical mass of users needed to evaluate social filtering techniques.

Another serious impediment to the large scale evaluation of social filtering techniques is the difficulty of constructing suitable measures of effectiveness. Recall, precision and fallout are of some use when comparing content-based filtering techniques, but their reliance on normative judgements of document relevance suppresses exactly the individual variations that social filtering seeks to exploit. One feasible evaluation technique would be to apply simulated users like those used by Sheth to investigate specific aspects of collaborative behavior. Impor-

tant issues such as the learning rates and variability in learning behavior across large heterogeneous populations could be investigated with large collections of simulated users whose design was tailored to explore those issues.

Another alternative is to study situated users (i.e., human users performing self-directed tasks), attempt to provide them with desirable documents, and then measure something related to their satisfaction. Those "dependent variables" could certainly be the sort of explicit feedback commonly required in present social filtering experiments, but insisting on explicit feedback increases the difficulty of assembling a sufficiently large user population. If suitable sources of implicit feedback can be identified, those same measures would be a far better choice for the set of dependent variables. Such an experiment design requires that separate training and evaluation document collections be used, a feature easily introduced by withholding implicit feedback from the filtering algorithm during the evaluation period. This approach can be used to evaluate both content-based and social filtering systems, so it would be a natural choice when evaluating systems which applied both types of techniques. It can only be applied, however, after suitable sources of implicit feedback are found. Since implicit feedback has the potential for a high payoff in performance evaluation, filtering effectiveness, and user satisfaction, research on that topic should be accorded a high priority.

2.7   Summary

The design of text filtering systems in general, and our work on adaptive multilingual text filtering systems in particular, benefit from research in text retrieval, user modeling and a number of other fields. Text filtering is, however, a unique

information seeking process that is distinguished by a focus on satisfying relatively stable interests in documents containing text. This chapter has reviewed progress in the field with particular emphasis on the selection component of the filtering process. Other useful perspectives are offered by Jiang [74], Mock [95], Stevens [143], and Wyle [159].

Text filtering systems must develop representations of both documents and user interests, they must be endowed with some way of comparing documents with interests, and they must possess some way of using the results of those comparisons to assist the user with document selection. Text retrieval research has produced a number of content-based representations that use the frequency with which terms appear in documents, and social filtering research has produced a complementary set of features based on shared annotations from other users. In this chapter we have described the representations that have been developed for a single language because (with the exception of the Fast Data Finder's multiple profile approach) those have been the only type of representations to be applied to text filtering. When combined with implicit or explicit feedback from the user about the documents they have examined, text representations provide a basis for construction of profiles which represent user interests.

Both text retrieval and machine learning offer techniques for comparing document representations with profiles, and this is an area of active research which we examine in considerable detail in Chapter 3. Document visualization is another dynamic research area, but ranked output presently offers a simple way of synergistically exploiting the strengths of human and machine to facilitate the filtering process so that is the approach we have chosen to exploit. In the next chapter we describe a specific technique for adaptively defining a profile and then

using that profile in a manner inspired by text retrieval practice to rank newly arrived documents. We then extend that technique to produce a user model capable of greater fidelity and conduct an experiment to determine whether this change results in improved effectiveness. The results of that experiment provide us with a key component for our comparison of adaptive multilingual text filtering techniques in Chapter 5.

Chapter 3

Gaussian User Model

This chapter describes a cognitive model for adaptive text filtering and presents experimental results comparing its performance to that of existing techniques. We have described this model previously in [103], and additional results are reported in [101]. Like the LSI-mean technique that we describe in detail below, we seek to exploit explicit relevance evidence and vector representations to predict the relevance of future documents with respect to a single interest. The unique feature of our approach, which we call the Gaussian User Model, is that we seek to determine whether improved performance can be achieved by accounting for observed differences in the importance to human readers of specific components of the vector representations. In this way, we seek to model an aspect of human cognition and to determine whether the greater potential fidelity of our model can be used to improve the effectiveness of an adaptive text filtering system.

The feature vectors we use are constructed using Latent Semantic Indexing (LSI), both because shorter vectors offer greater flexibility in the choice of machine learning algorithms and because (as we shall see in Chapter 4) Latent Semantic Indexing is also the basis for an interesting multilingual text retrieval

technique. At the end of this chapter we will use the experimental results reported in here to choose the user modeling technique that will serve as a basis for the adaptive multilingual text filtering experiments that are described in Chapter 5.

As Schütze, *et al.* have shown, LSI feature vectors and explicit feedback can be used to build a number of different types of profiles [132]. With one exception, the techniques they compared required that both positive and negative training examples be available in order to construct profiles. Since negative examples greatly outnumber positive examples in many text filtering applications, explicit feedback of negative examples would impose significant demands on the user. Identification of negative training examples through implicit feedback may well be practical (e.g., if a document title is displayed and that document is not selected, then it was not desired), but research on the use of implicit feedback is not yet sufficiently mature to justify such assumptions.

The one approach they used which did not require negative training examples was a technique which we refer to as the "LSI-mean" adaptive text filtering technique which was originally developed by Dumais [41].[1] In the LSI-mean technique the numerical average of the LSI feature vectors representing the relevant training documents is used as the profile. Documents are then ranked by decreasing cosine similarity with the profile in the same manner as when vector representations are used for text retrieval. In the Third Text Retrieval Conference (TREC-3), Dumais demonstrated good performance with this technique,

---

[1] This terminology is ours. Dumais refers to this technique as "LSI routing." Since several text filtering techniques exploit the LSI representation, we prefer the more descriptive name "LSI-mean."

achieving an average precision of 0.37.[2]. Furthermore, the LSI-mean technique outperformed an LSI text retrieval approach which achieved an average precision on 0.29 by forming the query from an explicit topic specification.

A useful perspective from which to view the LSI-mean technique is that the reduced-dimensional representations of the relevant training documents are samples of a random variable which has an unknown distribution and the mean is used as statistic to characterize that distribution. The results Dumais' reports in [41] thus suggest that (at least in TREC-3) with a reduced-dimensional vector representation the information need can usually be fairly well represented by the first moment of the distribution (the mean), and that the number of samples available (typically around 200 per topic) is sufficient to closely approximate that mean.

Dumais' results would be consistent with the hypothesis that the distribution is fairly strongly unimodal, tending to cluster around a single point without many interspersed non-relevant documents, but those results offer no insight into other characteristics of the distribution. A unimodal distribution for a narrowly defined topic in a vector space is intuitively appealing, since it provides both a mathematical and a geometric analogue to the clustering we imply when we speak of "nearby" and "distant" concepts, while allowing for the imprecision introduced by eliminating the effect of word order in the vector representation. Dumais' work leaves open the question of whether the symmetry inherent in the cosine measure can be improved upon, however. In this chapter we explore that question by considering techniques which estimate both the first and second moments of the distribution and then use those estimates to construct a rank

---

[2]TREC-3 "routing" results ranged from an average precision on 0.25 to 0.41

order which accounts for observed directional sensitivity in the interest evidence.

The second moment of a random vector's distribution is known as covariance. In a single dimension, variance is the expected deviation from the mean. In a vector space, covariance extends the concept of variance by representing the expected deviation from the mean in every possible direction. We use estimates of the mean and the covariance to rank previously unseen evaluation documents in order of decreasing likelihood that they are drawn from a distribution characterized by the estimated first and second moments. Because a Gaussian distribution is uniquely characterized by it's first and second moments, we call our technique the "Gaussian User Model."

We begin by describing the LSI-mean technique, including a detailed description of Latent Semantic Indexing and a brief recapitulation of Dumais' experimental results. We next describe the cognitive model which inspires our choice of covariance as a logical basis for extending Dumais' technique, and then present the Gaussian User Model in detail. With that as background, we describe an experiment comparing the Gaussian User Model with Dumais' technique and discuss our experimental results which show that the Gaussian User Model does not outperform Dumais' simpler LSI-mean technique. The Gaussian User Model is most similar to the discriminant analysis technique investigated by Schütze, *et al.*, so we next introduce their work and identify the similarities and differences. We conclude by selecting the LSI-mean technique as the basis for our multilingual experiments and by using our results to draw some fundamental conclusions about the nature of vector space user models for text filtering which are equally valid for monolingual and multilingual applications.

3.1 Latent Semantic Indexing

Latent Semantic Indexing (LSI) was originally developed as a vector space text retrieval technique [37]. The classic vector space method incorporates five types of semantic information:

- Removal of common low-content words using a "stoplist,"

- Rules for replacing words with suitable word stems,

- A term weighting scheme by which local (within-document) and global (across-document) information are used to alter vector component values,

- The cosine measure, which provides an operational definition for similarity, and

- Ranked output, which implies a monotone relationship between similarity to a query and the degree of relevance to the associated topic.

To these sources of semantic information, LSI adds a sixth:

- Substitution of shorter vectors in which semantic information is preserved but the effects of term usage variations are reduced.

LSI achieves this effect by constructing a linear mapping from the space spanned by a collection of vectors which describe documents to a reduced-dimensional subspace. As is the case for term weighting, LSI constructs this mapping by using global information in a prespecified manner. The two techniques are complementary, rather then competitive, however. Term weights are often computed by a nonlinear mapping which treats every term independently. LSI can compute only a linear mapping, but the values associated with several terms can

influence the result. Dumais has found that using term weighting with LSI produces better performance than when either technique is used in isolation [42].

Document Number	Sentence
1	. . . improvement over raw term matching . . .
2	. . . compared the performance of the latent structure . . .
3	. . . uses the estimation of latent structure . . .
4	. . . calculations have exact analogs in the latent model . . .

Table 3.1: Short example "documents" for LSI.

The key insight in LSI is that just as a document is represented by a vector of term frequencies, a term can be represented as a vector of document frequencies. This is easily seen in a simple example. Treating the short sentence fragments in Table 3.1 as documents for the purpose of this example, vectors constructed using the technique shown in Figure 1.1 are used as the columns of the term-document matrix in Table 3.2.[3] Just as document 1 is represented by the column vector $(0000001001001010100)^{\mathrm{T}}$, the term "latent" can be represented in this collection by the row vector (0111). This vector succinctly summarizes everything that is revealed about the term "latent" by the vectors which describe these four documents. Computing cosine similarities, for example, we see that "latent" is used in a manner similar to "the" (cosine=0.97) and fairly similar to "structure" (cosine=0.90). This somewhat contrived example is intended only to illuminate the sense in which vectors can represent terms as well as documents.

---

[3]The sentence fragments are drawn from the first paper on LSI [54]. No stemming or term weighting has been applied, and no stopwords have been removed in this example.

75

The foundation of LSI is that for larger document collections the correspondence between semantic similarity and usage pattern similarity is sufficiently strong to automatically extract semantic information from these patterns.

| Term | Document Number | | | |
	1	2	3	4
analogs	0	0	0	1
calculations	0	0	0	1
compared	0	1	0	0
estimation	0	0	1	0
exact	0	0	0	1
have	0	0	0	1
improvement	1	0	0	0
in	0	0	1	0
latent	0	1	1	1
matching	1	0	0	0
model	0	0	0	1
of	0	1	1	0
over	1	0	0	0
performance	0	1	0	0
raw	1	0	0	0
structure	0	1	1	0
term	1	0	0	0
the	0	2	1	1
uses	0	0	1	0

Table 3.2: An example term-document matrix.

LSI is, in essence, an automatic technique for recognizing similarities in the way terms are used in a collection and then suppressing the effect of term usage variations in order to conflate similar terms towards a "conceptual" representation. In fact, the output of LSI is a "T_0" matrix which, like the term-document matrix, has one row for each term in the collection. The rows of the T_0 matrix contain row vectors which represent the terms in such a way that terms with

similar usage are assigned similar vectors, while terms with significantly different usage are assigned dissimilar vectors. For example, the terms in Table 3.2 are represented in LSI with the four-dimensional vectors in the T_0 matrix shown in Table 3.3.

Term	Dimension			
	1	2	3	4
analogs	0.11	0.39	0.00	0.07
calculations	0.11	0.39	0.00	0.07
compared	0.18	-0.10	0.00	-0.39
estimation	0.15	-0.16	0.00	0.43
exact	0.11	0.39	0.00	0.07
have	0.11	0.39	0.00	0.07
improvement	0.00	0.00	0.45	0.00
in	0.15	-0.16	0.00	0.44
latent	0.44	0.13	0.00	0.12
matching	0.00	0.00	0.45	0.00
model	0.11	0.39	0.00	0.07
of	0.33	-0.26	0.00	0.05
over	0.00	0.00	0.45	0.00
performance	0.18	-0.10	0.00	-0.39
raw	0.00	0.00	0.45	0.00
structure	0.33	-0.26	0.00	0.05
term	0.00	0.00	0.45	0.00
the	0.63	0.02	0.00	-0.27
uses	0.15	-0.16	0.00	0.44

Table 3.3: The LSI T_0 matrix for the term-document matrix in Table 3.2.

The T_0 matrix has two useful properties:

1. Any pair of document representations that are both formed as a particular linear combination of the rows of the T_0 matrix will have exactly the same cosine as the corresponding columns of the original term-document matrix.

2. Removing the last few components of each vector in the matrix T_0 matrix will often significantly improve the rank ordering that is produced when this technique is used to compute the cosine similarity measure between a profile (or query) vector and vectors representing the documents.

The first property is easily illustrated with a simple example. The cosine of the vectors for documents 2 and 3 in Table 3.2 is 0.79. Table 3.4 shows how each each row of Table 3.3 is multiplied by the weight of the associated term in column 2 of Table 3.2 and then summed to produce the short vector $(2.73, -0.56, 0, -1.12)$ which represents document 2. Repeating the process using term weights from column 3 of Table 3.2 produces a representation of $(2.17, -0.85, 0, 1.25)$ for document 3. The cosine of the angle between these two vectors is 0.79, verifying property 1 for this example.[4]

Term	weight	Dimension			
		1	2	3	4
compared	1	0.18	-0.10	0.00	-0.39
latent	1	0.44	0.13	0.00	0.12
of	1	0.33	-0.26	0.00	0.05
performance	1	0.18	-0.10	0.00	-0.39
structure	1	0.33	-0.26	0.00	0.05
the	2	0.63	0.02	0.00	-0.27
Linear Combination		2.73	-0.85	0.00	1.25

Table 3.4: Calculation of the LSI feature vector describing document 2.

[4]In fact this result is even stronger because it is actually the inner product of any two vectors which is preserved. As a result, the cosine in both cases can be computed as $\sqrt{\frac{5}{\sqrt{9}\sqrt{7}}}$ using equation (2.5).

This first property establishes the full ranking effectiveness of the LSI at that achieved by the classic vector space method. But the more remarkable property is that removing the last few components of each vector in the matrix T_0 (producing a matrix we call T) can actually improve the effectiveness of a rank ordering system. Figure 3.1 shows the variation in average precision as the number of dimensions retained increases for a standard information retrieval test collection using two different term weighting functions.[5] The upper plots were produced using the "ltc" weights computed using the following four steps:[6]

$$\text{local weight} \quad = \quad \ln(\text{occurrences of term } i \text{ in doc } k) \qquad (3.1)$$

$$\text{collection-wide weight} \quad = \quad \ln(\frac{\text{number of docs}}{\text{number of docs with term } i}) \qquad (3.2)$$

$$\text{unnormalized weight} \quad = \quad \text{local weight } * \text{ global weight} \qquad (3.3)$$

$$\text{normalized weight} \quad : \quad \frac{\text{unnormalized weight}}{\sqrt{\sum_i (\text{unnormalized weight}_i)^2}} \qquad (3.4)$$

The upper horizontal line shows the average precision achieved using the original vectors of term frequencies, and the line with the pronounced peak represents the average precision achieved when the "ltc" term weights are used with LSI. LSI does not always exceed the performance of the classic vector space method, however. The lower curve in Figure 3.1 represents the performance of LSI when raw term frequency with no collection-wide adjustments and no cosine normalization is used, and the lower horizontal line shows the average precision achieved with

[5] These results were obtained using the Cranfield collection of 1398 aerospace abstracts that we describe in Section 3.3, and are averaged over 225 queries.

[6] The notation we use for term weighting functions is drawn from the SMART experimental text retrieval systemthat we describe in Section 3.3. The first letter specifies the local weight, the second the collection-wide weight, and the third the normalization technique. The "ltc" nomenclature is read as "log," "tfidf," "cosine" weighting.

the original vectors under those conditions. With these "nnn" weights,[7] LSI consistently reduces the quality of the rank ordering (unless almost every dimension is retained), and no choice of dimensions actually improves performance.
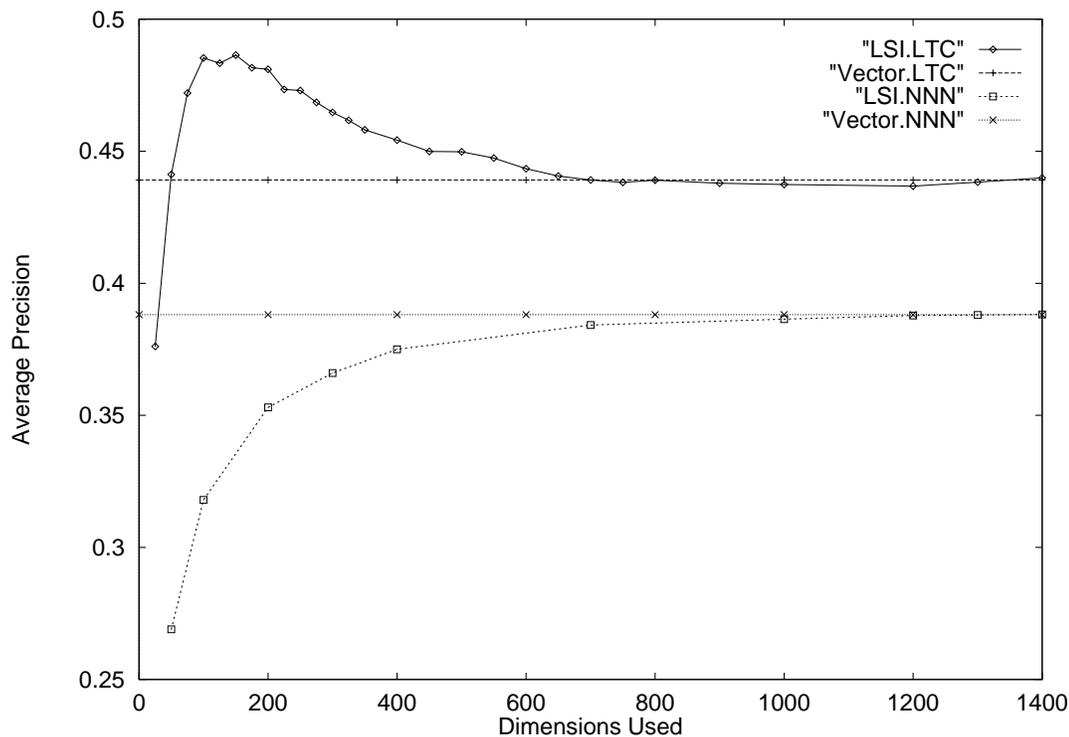
Figure 3.1: Effect of varying the number of retained dimensions on LSI effectiveness.

Although LSI does not always improve performance, examples like the "nnn" case above in which LSI significantly reduces the quality of a rank ordering are actually fairly rare. When significant improvements do occur, the best explanation for that improvement is that removing the last few dimensions generally moves terms with similar meaning closer together (i.e., their representations become more similar), but that terms with dissimilar meanings remain far apart

[7]The SMART "nnn" designation is read as "no adjustment to term frequency, no collection-wide weight adjustments and no normalization.

in the lower dimensional space. Just as a human might choose to subsume two slightly different terms under the heading of a broader term when constructing a thesaurus, in LSI the representation of terms which have similar usage (in the training collection) is compressed towards a single combined representation when the last dimensions are removed.

As a somewhat extreme (but easily seen) example, examining only the first dimension (column 1) in Table 3.3 would lead to the conclusion that the usage of "latent" should be fairly similar to the usage of "of," and that the usage of "analogs" should be quite different from the usage of "the." A glance at Table 3.2 shows that this accords well with the similarities that would be computed using the patterns in the original term-document matrix. Larger collections provide more fine-grained term representations and larger numbers of retained dimensions provide a richer variety of ways in which documents can be represented in the reduced-dimensional space, but the key idea of moving similar objects closer together while preserving the differences between dissimilar objects is the same. It is the demonstrated ability of LSI to abstract such representations which encode "latent" semantic meaning from the structure of the term-document matrix that led Deerwester, *et al.* to coin the name Latent Semantic Indexing [37]. In the next section we describe the mathematical details of the technique and discuss the way in which the number of dimensions to be retained is determined.

3.1.1 Mathematical Details

The mathematical basis for LSI is the Singular Value Decomposition (SVD) of the term-document matrix. The SVD identifies a useful set of basis vectors for the column space of a term-document matrix. These basis vectors span the

same space as the collection of vectors which represent documents and, because they are computed by a rotation (multiplication by an orthogonal matrix) of the original vectors, applying the change of basis leaves the inner product of any pair of vectors unchanged. This is the characteristic which underlies the first property of the T_0 matrix. The second property described on page 78 results because the basis vectors chosen by the SVD are chosen and sorted in such a way that removal of one or more from the end of the sorted set will induce the minimum possible average change to the unnormalized inner product of two vectors from the collection. In other words, if r basis vectors suffice to span the space and the last n basis vectors are removed from the sorted set produced by the SVD, the mean magnitude of the change in the set of possible inner products will be the minimum that could be achieved in any subspace with $r - n$ basis vectors. In the remainder of this section we formalize that notion.

From a collection of documents, a term-document matrix X is formed. Each entry in X consists of a single term weight for the term associated with its row and the document associated with its column, computed with any appropriate function of within-document and across-document term frequencies. The SVD of this matrix is then computed to find:

T_0 : A matrix with orthonormal columns of "left singular vectors" which span the column space of X,

D_0 : A matrix with orthonormal columns of "right singular vectors" which span the row space of X, and

S_0 : A diagonal matrix of singular values which accumulates the results of normalizing T_0 and D_0,

such that $X = T_0 S_0 D_0^{\mathrm{T}}$. This decomposition is illustrated in Figure 3.2. By convention, the left and right singular vectors are constructed so that the values on the main diagonal of S_0 will be nonnegative and sorted in decreasing order. For a matrix X of rank r, the S_0 matrix will have exactly r positive (i.e., nonzero) entries. The existence (and conditions for the uniqueness) of the SVD of any matrix is easily shown (c.f., [145]). We compute the SVD using the single vector Lanczos method, a standard numerical analysis technique adapted for sparse matrices by Berry, *et al.* [9].[8]
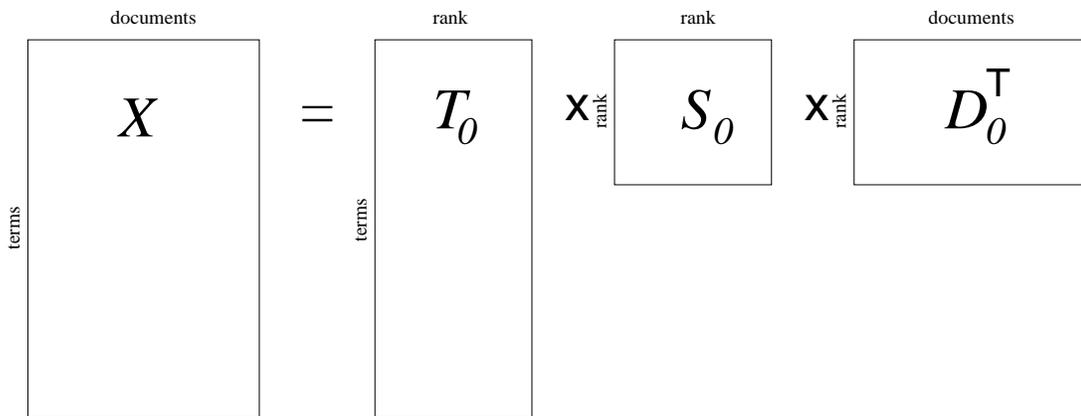
Figure 3.2: Singular Value Decomposition of the term-document matrix X.

Since S_0 is diagonal, the SVD of X can be rewritten as:

$$X = \sum_{i=1}^{r} T_{0*,i} S_{0i,i} D_{0*,i}^{\mathrm{T}} \tag{3.5}$$

where $T_{0*,i}$ (the ith left singular vector), S_{0ii} (the ith singular value) and $D_{0*,i}$ (the ith right singular vector) are together referred to as the ith singular triple.

---

[8]In particular, we use a version of the the las2.c file from SVDPACKC which we have modified to conform to the calling conventions used in the SMART text retrieval system. SVDPACKC is available from http://www.netlib.org/svdpack and SMART is available from ftp://ftp.cs.cornell.edu/pub/smart. Details of our modifications can be found in Appendix A.

The key property of the SVD for LSI is that because the singular values are sorted in descending order the best possible rank k approximation to X (for any rank $k < r$, in the least squares sense) can be computed as:

$$\hat{X} = \sum_{i=1}^{k} T_{0*,i} S_{0i,i} D_{0*,i}^{\mathrm{T}} \qquad (3.6)$$

In other words, removing the last $r - k$ singular triplets results in the smallest possible change to the average inner product of any two columns of X. Figure 3.3 depicts the matrices T, S and D that result from retaining the singular triples associated with the k largest singular values, and Tables 3.5 and 3.6 summarize the notation that we use in this section.
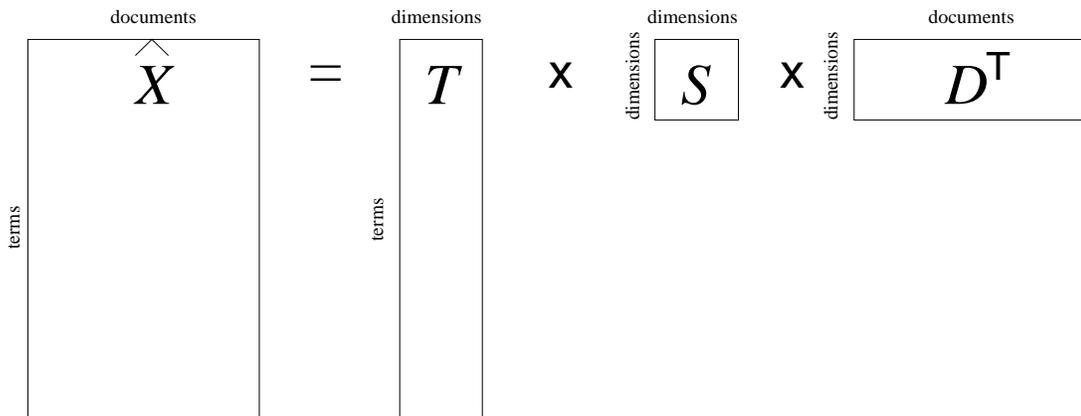
Figure 3.3: Result of retaining k singular triples.

Quantity	Interpretation
t	Number of terms
d	Number of documents
r	Rank of the term-document matrix
k	Number of retained dimensions

Table 3.5: Summary of SVD matrix dimensions.

Matrix	Dim	Contents
X	$t \times d$	Original term-document matrix
T_0	$t \times r$	Orthonormal columns are left singular vectors of X
S_0	$r \times r$	Diagonal elements are the singular values of X
D_0	$d \times r$	Orthonormal columns are right singular vectors of X
T	$t \times k$	Retained left singular vectors
S	$k \times k$	Retained singular values
D	$d \times k$	Retained right singular vectors
$\hat{X}$	$t \times d$	Approximate term-document matrix formed as TSD^{T}

Table 3.6: Summary of SVD matrix notation.

The effectiveness of LSI depends on the ability of the SVD to extract salient features from the term frequencies across a set of documents. To understand this behavior it is helpful to develop an operational interpretation of the three matrices which make up the SVD. In the original vector space representation, $X^{\mathrm{T}}X$ is a $d \times d$ symmetric matrix of inner products between vectors representing documents (which we call "document vectors"). Each column of the $X^{\mathrm{T}}X$ matrix is a set of inner products between the document vector in the corresponding column of X and every document in the collection. The cosine similarity measure for documents i and j can be computed as:

$$\frac{(X^{\mathrm{T}}X)_{i,j}}{(X^{\mathrm{T}}X)_{i,i} \cdot (X^{\mathrm{T}}X)_{j,j}} \tag{3.7}$$

Expanding X using the SVD,

$$X^{\mathrm{T}}X = D_0 S_0 T_0^{\mathrm{T}} T_0 S_0 D_0^{\mathrm{T}} \tag{3.8}$$

$$= D_0 S_0 S_0 D_0^{\mathrm{T}} \tag{3.9}$$

$$= (D_0 S_0)(D_0 S_0)^{\mathrm{T}} \tag{3.10}$$

Considering each column individually,

$$X_{*,i}^{\mathrm{T}}X_{*,j} = (D_0S_0)_{i,*}(D_0S_0)_{j,*}^{\mathrm{T}}\forall i,j \tag{3.11}$$

In other words, the cosines computed with columns of X will be identical to the cosines computed using the rows of D_0S_0.

Given any document vector $X_{*,i}$, $(D_0S_0)_{i,*}$ can be calculated as $X_iT_0^{\mathrm{T}}$, since $X = T_0S_0D_0^{\mathrm{T}}$ leads directly to

$$T_0^{\mathrm{T}}X = T_0^{\mathrm{T}}T_0S_0D_0^{\mathrm{T}} \tag{3.12}$$

$$= S_0D_0^{\mathrm{T}} \tag{3.13}$$

Together with equation (3.11), this proves our first claim about the LSI, that we can produce a dense vector of rank r which will produce the same cosines as the sparse rank t document vectors, and that that dense vector can be formed as the linear combination of the rows of the T matrix that is specified by the original document vector.

Our second claim is that eliminating the last several columns of T will often improve the representation. Of course, (as Figure 3.1 illustrates for the case of "nnn" term weights) this is not always true. So rather than prove our second claim we instead formalize the intuitive explanation of why an improvement in ranking performance is often observed that we introduced above. Our explanation is motivated by the approach suggested by Deerwester, *et al.* [37], but the details of our presentation are original.[9]

In the computation of the cosine similarity measure in the vector space model, each element of the feature vector is treated identically. In other words, the

[9]Alternative explanations of the effectiveness of LSI which are based on the same concepts have recently been offered by Story [144].

vector space model treats terms that happen to have similar meaning in exactly the same way that it treats unrelated terms. Elimination of the small singular values from S_0 amounts to a judgment that the features associated with small singular are harmful when representing terms. To see why this is so, it is helpful to examine how the SVD affects the vectors which represent terms.

Recall that each row of X represents all of the information that X encodes about a term. Repeating the derivations starting from XX^T, we find that cosines computed with rows of X are identical to cosines computed using the rows of $T_0 S_0$ and that

$$XD_0 = T_0 S_0 \tag{3.14}$$

Since each value in S selects (and scales) a single column in T, eliminating small singular values amounts to eliminating the columns of T which have the smallest average effect on term similarity computations. Furnas has demonstrated that humans often use a variety of words to describe the same concept [53] and Deerwester, *et al.* suggest that retaining only the singular vectors associated with the largest singular values captures the underlying semantic structure (i.e. the concepts) in the term-document matrix while rejecting the "noise" that results from term usage variations [37]. In other words, the elimination of the small singular values reduces the document feature space into a "document concept space."

This analysis motivates the description of T^T as a linear function from a document vector (which specifies which terms appear in the document) to a "concept vector" which encodes the concepts which appear in a document. Similarly, SD^T represents a linear function from a term vector (which specified which documents a term appears in) to the "concept vector" which represents a

term. Table 3.7 depicts this relationship.

Matrix	Row	Column
T	Concept Vector Space	Term Vector Space
DS	Concept Vector Space	Document Vector Space

Table 3.7: Spaces spanned by the left and right singular vectors.

Although we are able to describe the effect of eliminating the smallest singular values, our explanation offers little insight into the number of singular values which should be retained. Inspection of the singular values for the Cranfield collection in Figure 3.4 reveals that the pronounced peak in Figure 3.1 appears to be associated with the departure from linear decay of the singular values on a log–log plot, but is not known whether a similar cue exists for other collections or term weighting strategies.[10] Results reported by Dumais do, however, suggest an empirical correlation between the breadth of a test collection (i.e., the number of topics addressed in the collection) and the optimal value of k. On a collection of 1,033 medical abstracts, Dumais found that average precision was maximized at between $k = 70$ and $k = 100$ [42] and Figure 3.1 shows a similar maximum between $k = 100$ and $k = 150$ on a collection of 1398 aerospace abstracts. But for a larger collection containing 38,175 news articles, performance continued to improve up to $k = 346$, the largest value that was tried [41]. This "what works best" approach is presently the best known approach for choosing k, with values between $k = 100$ and $k = 400$ typically found to offer a reasonable performance without imposing exceptional computational demands [10].

---

[10]This observation was originally made by Dr. Christos Faloutsos. SMART "ltc" term weights were used to produce this plot.
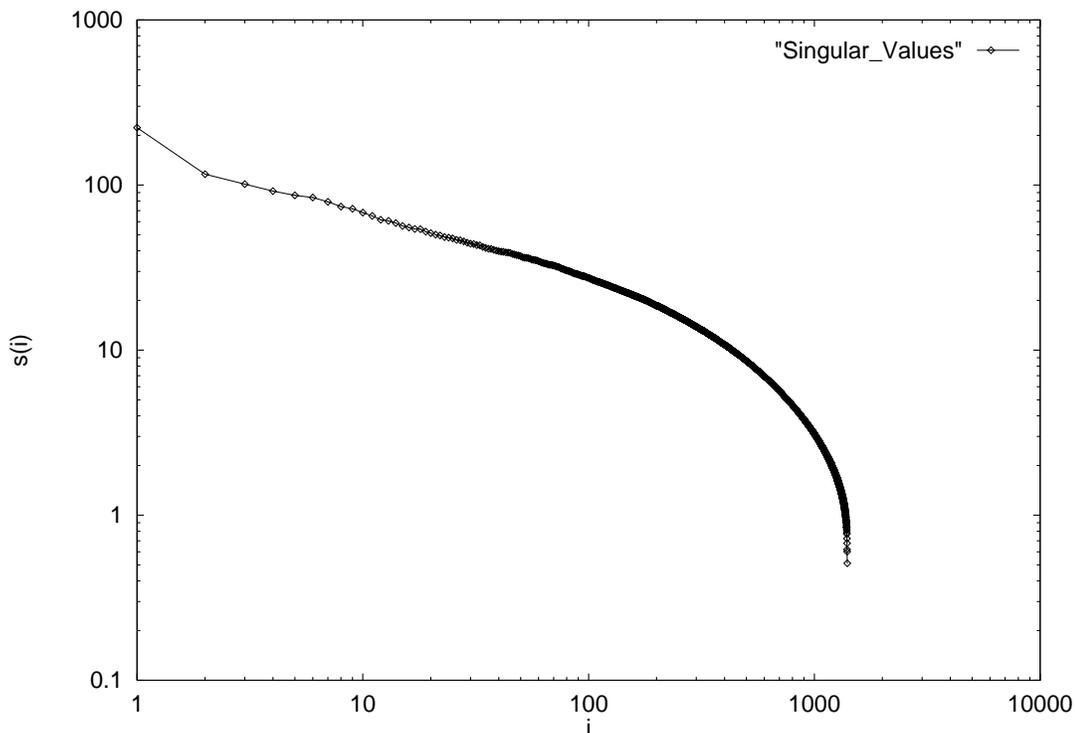
Figure 3.4: Singular values for the Cranfield collection (log–log).

The sparse matrix Lanczos method we use is an iterative algorithm that is well suited to computing the singular triples associated with the largest singular values. Typically the largest singular triples can be found in time proportional to the number of documents, but the time required for the computation increases with the square of the number of singular triples that are desired. Thus, large document collections pose no particular problems by themselves, but the larger numbers of singular triples needed to achieve good performance on those collections does pose a difficulty. The time complexity of the algorithm is:

$$4in + 2kn + i^2d \tag{3.15}$$

where n is the number of of nonzero elements in X, i is the number of iterations that are required to compute the desired number of singular values and k and d

89

are defined in Table 3.5 [7].[11]

In text retrieval applications it is reasonable to treat the number of terms in any document as an independent and identically distributed random variable. Thus as d becomes large, n will be well approximated by a constant multiple of d. The number of iterations is upper bounded by the rank of X (which is in turn upper bounded by $\min\{d, t\}$), but in practice it can be fairly closely approximated by a constant multiple of k when $k \ll \min\{d, t\}$, as Table 3.8 shows for the Cranfield collection.

Documents (d)	Dimensions (k)		
	100	200	300
200	176	200	N/A
400	212	332	400
600	224	370	484
800	226	384	514
1000	236	392	526
1200	234	396	540
1398	232	409	550

Table 3.8: Number of iterations required to compute k dimensions using SVD on the Cranfield collection.

Thus it is the $i^2 d$ term in equation (3.15) that dominates the asymptotic time complexity of the sparse matrix SVD calculations in text retrieval applications when the Lanczos method is used, and the resulting asymptotic time complexity is $k^2 d$. Table 3.9 shows timing results for SVD computations on term-document matrices formed from the Cranfield collection on a SPARC 20.

---

[11]The equation presented in [7] omits the third term, which is only required when the singular vectors are desired, and it includes some smaller terms that are dominated by the terms shown here.

Dimensions (k)	Documents (d)					
	400	600	800	1000	1200	1398
100	0:28	0:41	0:56	1:09	1:19	1:29
300	2:59	6:25	9:10	10:06	11:57	14:27

Table 3.9: Time to compute the SVD for the Cranfield collection (min:sec).

The space complexity of the algorithm is somewhat more tractable. The space required is proportional to [106]:[12]

$$2ki + i^2 + 2t + 1.5n \qquad (3.16)$$

As more documents are added to the collection, t (the number of unique terms) will eventually grow more slowly than d. So initially equation (3.16) will be dominated by $k^2 + t$, but for very large collections it will eventually be dominated by $k^2 + d$. So although the asymptotic space complexity of the algorithm is also proportional to the number of documents and to the square of k, for the asymptotic space complexity the combination is additive rather than multiplicative. Since we do not yet know whether the optimum value of k grows more or less slowly than $\sqrt{d}$, it is not presently possible to determine which of these two terms will dominate the asymptotic space complexity.

3.1.2 The LSI-Mean Filtering Technique

In Chapter 2 we briefly described how both Foltz and Schütze, et. al. have used LSI feature vectors with machine learning techniques [48, 132]. Dumais has used LSI in conjunction with "relevance feedback," an information retrieval technique in which training documents are used to construct a single representation of an

---

[12]Again, less significant terms are not shown.

interest [41]. That approach is distinguished from other applications of relevance feedback to vector representations by two characteristics:

- The relevance feedback is done in the reduced-dimensional LSI space.

- Only the relevant training documents are used to construct the profile.

In Dumais' LSI-mean adaptive text filtering technique, the SVD and small singular value rejection steps are first performed using a representative sample of the relevant and non-relevant documents in the training set. Concept vectors for the documents judged to be relevant to each topic are then computed. The mean of the relevant document concept vectors is computed for each topic and that mean vector is used as the profile. Documents in the evaluation set are then rank ordered by decreasing cosine between the profile and the evaluation documents' concept vectors.

In the TREC-3 "routing" evaluation, Dumais computed the SVD on the approximately 78,746 word stems which occurred in five or more of the 38,175 documents in the training collection for which relevance judgements were available, using the "ltc" term weight formula.[13] Profile vectors for fifty topics were then computed based on between 25 and 742 (mean 215) relevant documents per topic. Retaining 346 dimensions, Dumais then computed concept vectors for 336,306 documents in the evaluation collection [41]. The average precision of 0.3737 achieved using this technique greatly exceeded the 0.2880 average precision achieved when the concept vector for the topic specification was substituted

---

[13]TREC includes both a "routing" evaluation and a "filtering" special interest track. The "routing" evaluation evaluates ranked output systems, while the "filtering" evaluation evaluates exact-match systems.

for the profile, and is close to the 0.4068 average precision achieved by the best participating system.

3.2 The Gaussian User Model

Dumais' results suggest that a statistical user model can be effective, even when a very simple user model is used. The LSI-mean model represents each information need with a single vector and then ranks documents by how "close" they are to that representation, regardless of the "direction" of the difference. The "direction" may, however, encode important information as well because LSI produces vectors which behave in some ways as if they represent concepts. For this reason, we have investigated the performance of a technique which extends the LSI-mean model with an asymmetric similarity measure that we first described in [103].

3.2.1 A Cognitive Model for Document Selection

By a "cognitive" model we mean one which approximates in some way a high-order human decision-making process. The process we wish to model is that of sorting newly arrived documents into a list ranked by their degree of relevance to an information need. Our goal is to use vector representations of documents and positive training examples to order the newly arrived documents in a way that is as similar as possible to that which would be arrived at by a human cognitive process. The LSI-mean technique has demonstrated good performance in this same task, so we have chosen to extend that technique to model an aspect of human cognition.

The aspect of human cognition we seek to model is "specificity of interest." Our intuition suggests that humans treat a relatively small number of concepts with great specificity when identifying relevant documents (i.e., they must specifically be present in the right amounts, or they must specifically be absent), while other concepts are allowed a greater range of variation. We refer to this later case as "don't care" concepts, although even in this case humans may prefer that such concepts do not completely dominate the document. The cognitive basis for the Gaussian User Model is our belief that representations of human interests could benefit from an ability to assign differing tolerance for deviation from "typicality" for different concepts.

A simple example may help to visualize this approach. Consider a vector representation of two information needs using three conceptual dimensions that are developed from occurrences of evidence for the concepts "valuable," "green," and "paper." In order to motivate the mathematical techniques we describe below, suppose that we were to represent an interest in United States paper currency with approximately equal measures of all three concepts and that we chose to represent an interest in emeralds with approximately equal measures of "green" and "valuable," regardless of how often "paper" appears.

Because only positive evidence is used for each concept, three dimensional vectors that have been normalized to unit length will all appear on the surface of the first octant of a unit-radius sphere. The circle in Figure 3.5 describes the points at which individual vectors from the cone of vectors which contain approximately equal proportions of all three concepts (reasonable representations for documents about paper money) will meet the unit sphere. The vector with exactly equal proportions of the three concepts is shown at the center of the

circle. This corresponds to the LSI-mean profile. The LSI-mean approach is well suited to this situation because the cosine similarity measure is sensitive only to the amount of angular difference between the profile and the document vectors. So use of the cosine similarity measure is equivalent to ranking documents in order of increasing arc distance from the profile vector to the projection of the documents' concept vectors onto the unit sphere.
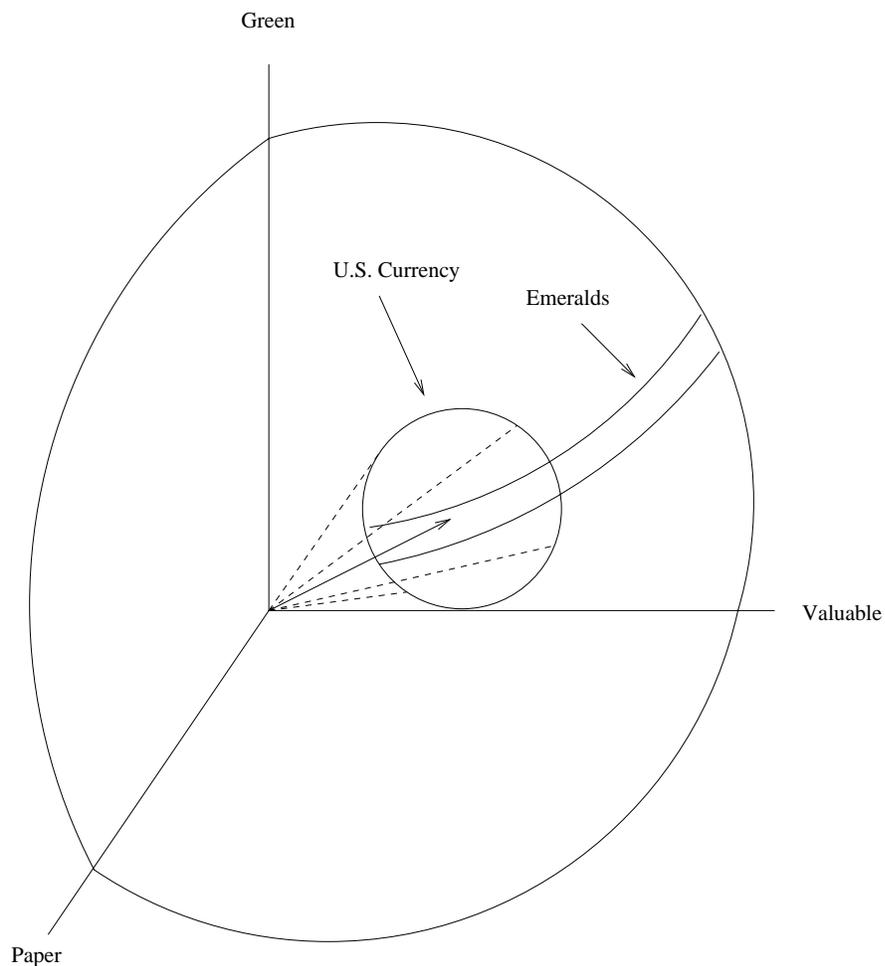
Figure 3.5: Circle and band interest representations on the unit sphere.

The band in Figure 3.5 describes the intersection of the group of vectors with approximately equal proportions of "green" and "valuable" without an

95

overwhelming proportion of "paper" (reasonable representations for documents about emeralds) with the unit sphere. This situation in which there is a "don't care" dimension (paper), is less well modeled by the LSI-mean approach. With any single profile vector and a rotationally invariant ranking technique, some documents outside the band will necessarily be ranked ahead of some documents that are inside the band, regardless of where that profile vector is placed. It is this limitation which our extension to the LSI-mean technique is designed to overcome.

In this example our simple cognitive model is known *a priori*, having been constructed by hand to illustrate these ideas. For adaptive text filtering it is also necessary to provide a means of learning the cognitive model. In this section we describe a directionally sensitive user model which is based on the same two adaptation characteristics as Dumais' LSI-mean technique:

- The relevance feedback is done in the reduced-dimensional LSI space.

- Only the relevant training documents are used to construct the profile.

3.2.2   Gaussian User Model Design

When ranked output is desired, the circle and band in Figure 3.5 can be thought of as prototypes for contours of equal rank on the surface of the sphere. Increasingly larger circles (or bands) would encompass increasingly larger number of documents. The first to be encompassed should be ranked the highest and the last ranked the lowest. This defines a concept of distance which is distorted by the shape of the prototype. The concept of a distorted distance measure on the unit sphere (or, in higher dimensions, unit hypersphere) is the fundamental idea

behind our extension to the LSI-mean technique.

In LSI, the SVD computation rotates the axes of the concept space in order to produce a diagonal S matrix. So when LSI is used, it is not practical to identify a single concept with each axis in the manner depicted in Figure 3.5. LSI-based techniques which account for varying levels of concept-specificity must therefore learn these models without obtaining human guidance on a concept-by-concept basis. Furthermore, the information need representation must be able to encode the level of specificity in arbitrary directions.

These considerations led us to consider characterizing concept specificity using both the mean and the covariance of the relevant training documents. This is a quite straightforward extension of the LSI mean approach. If the training documents are viewed as instances of a random vector, the LSI-mean user model represents the interest by estimating the first moment of the distribution (the mean) and then using a symmetric function to rank order newly arrived documents. A natural next step is to consider a model based on estimates of the first and second moments of the distribution and a rank ordering scheme which accounts for the differences revealed by the second moment of the distribution. Thus, estimates of the mean and the covariance of a random vector are the basis for the technique we call the the "Gaussian User Model."

It is well known that length normalization significantly improves the performance of text retrieval systems [51]. For example, normalizing each vector to unit length before computing the SVD significantly improved the performance of the LSI-mean technique in the results depicted in Figure 3.1. This makes sense intuitively as well, since human evaluation of the topical similarity of two documents should be more closely related to the relative distribution of concepts

in the two documents than to the actual number of references to each concept in each document. For example, a document formed by concatenating two copies of a shorter document would likely be judged by a user to have identical content. Singhal, *et al.* have recently shown that in some situations it can make sense (and improved retrieval performance) to make some adjustments for document length when selecting documents because in some applications longer documents typically address a larger number of topics [138]. We are not aware of a case in which this approach has been applied in conjunction with LSI feature vectors, however, so we chose to avoid this problem by using a fairly homogeneous document collection consisting of abstracts of similar length in the experiments we report below. This allows us to simply normalize each vector that represents a relevant document to unit length before using it to construct the Gaussian User Model.

The most straightforward way to address this problem would be to calculate a covariance-sensitive distance measure on the surface of the unit hypersphere (or, equivalently, to distort the unit hypersphere itself using the covariance matrix and then use a symmetric distance measure on the distorted manifold). But a more efficient approach is to project every concept vector to the hyperplane which is tangent to the mean vector and then compute what is known as "Mahalanobis distance" in that hyperplane [121]. Mahalanobis distance essentially allows us to rank order documents in order of increasing surprise that they would have come from the distribution represented by the mean vector and the covariance matrix.

Together, the mean vector and the covariance matrix uniquely specify a multidimensional Gaussian distribution. For such a distribution, the surfaces of

equal probability density are hyperellipses, and the Mahalanobis distance is simply the inverse of the probability density at which the deterministic vector is found. In directions with small variance, small Euclidean distances equate to large Mahalanobis distances. For directions with large variances, even large Euclidean distances will equate to relatively small Mahalanobis distances. If the variance is the same in every direction, Mahalanobis distance is simply a scaled version of the Euclidean distance. To the extent that a distribution is well characterized by its first and second moments, Mahalanobis distance is essentially a measure of our "surprise" at encountering a specific instance of a random vector.

Because the first and second moments of a random vector uniquely describe a Gaussian distribution, the user model described here is a "Gaussian User Model" in the sense that it implements a Gaussian distribution which seeks to approximate the true distribution more closely that would be possible with the rotationally invariant LSI-mean model. Figure 3.6 depicts the same example described above when ellipses depicting contours of constant probability density are substituted for the bands and circles. In this case the circle appears exactly the same, since a circle is simply a special case of an ellipse. The ellipse representing the band is a less perfect approximation, but it does capture some of the information about the region in which desirable vectors will be found.

As can be seen in Figure 3.6, a rank ordering of vectors produced using nested ellipses that represent contours of constant Mahalanobis distance in the hyperplane normal to the mean vector will be the same as that which would be produced with computations on the surface of the unit hypersphere, except for extremely distant vectors. Vectors which would violate this monotonicity property are easily recognized because they have a negative projection on the
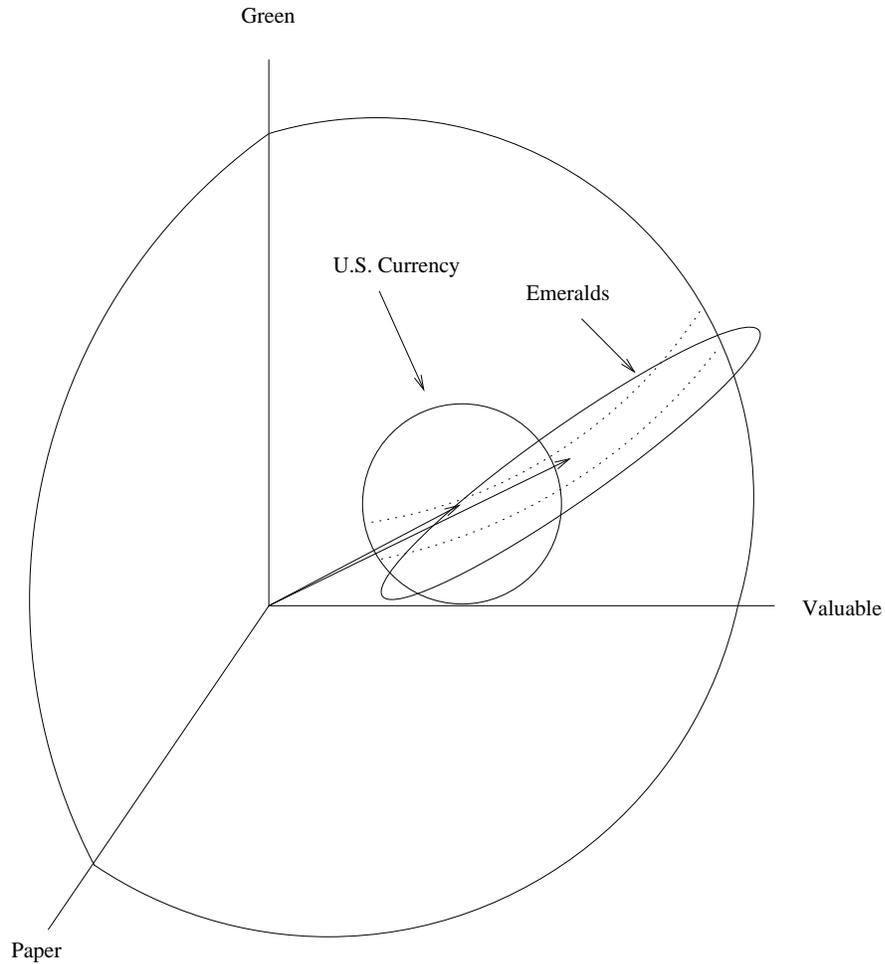
Figure 3.6: Contours of constant Mahalanobis distance on planes tangent to the surface of the unit sphere.

mean vector, and they can safely be grouped (unranked) at the bottom of the ranked list because that is where computations on the unit hypersphere would have placed them.

In the next section we present the mathematical details of the Gaussian User Model. The sections which follow present the results of an experiment which compares the performance of the Gaussian User Model with the LSI-mean technique and discuss the implications of those results.

3.2.3 Mathematical Details

We begin by estimating the mean vector and then projecting each document to the hyperplane that is normal to the mean vector. As in the LSI mean model, the sample mean of N relevant training concept vectors $\{v_1, v_2, \ldots, v_N\}$ is computed as:

$$\mu = \frac{1}{N} \sum_{i=1}^{N} v_i \tag{3.17}$$

The projection p_i of every vector v_i to the hyperplane normal to μ is then computed using Gauss-Seidel elimination as:

$$p_i = v_i - (v_i^{\mathrm{T}} \mu_i)\mu \tag{3.18}$$

This projection requires about twice as many elementary arithmetic operations as length normalization, but it enables the use of linear, rather than angular, measures for ranking.

Once the projections of the relevant documents are known, the sample covariance matrix of those projections can be formed as:

$$\Sigma = \frac{1}{N} \sum_{i=1}^{N} p_i p_i^{\mathrm{T}} \tag{3.19}$$

If this sample covariance matrix were a good estimate of the true covariance matrix, the Mahalanobis distance m from the distribution represented by μ (the projection of which is zero) and Σ to any projected vector p_i is defined as:

$$m^2 = p_i^{\mathrm{T}} \Sigma^{-1} p_i \tag{3.20}$$

Mahalanobis distance is a natural replacement for Euclidian distance in this case because both are computed as a quadratic form of the projection of a vector to a hyperplane. When Σ is the identity matrix, m in equation (3.20)will simply

be the Euclidian distance. Otherwise, for directions in which high variance is expected the effect of Σ^{-1} in the quadratic form will be to make the Mahalanobis distance smaller than the Euclidian distance would have been. Similarly, for directions in which small variance is expected the Mahalanobis distance will be larger than the Euclidian distance.

Unfortunately, two problems arise with this approach. First, the sample covariance may not have full rank, so Σ^{-1} may not exist. In particular, Σ is the sum of N rank 1 matrices, so the rank of Σ is upper bounded by N. Until there are more relevant training documents than dimensions, Σ is guaranteed not to have a unique inverse. A second difficulty persists even after Σ achieves full rank, however. Spectral decompositions of typical sample covariance matrices formed with relatively small numbers of samples (up to about three times the number of dimensions) into an orthogonal eigenvector matrix U and a diagonal eigenvalue matrix Λ usually reveal that the largest eigenvalue of the sample covariance matrix is too large and the smallest eigenvalue is too small [52]. Of course, the zero eigenvalues which occur before the sample covariance matrix achieves full rank are clearly too small, so the first problem is actually a special case of this second.

Friedman has proposed a technique known as "regularization" which addresses both difficulties [52]. Friedman suggests that by replacing every eigenvalue with a linear combination of that eigenvalue and the mean of the set of eigenvalues, this observed bias towards extreme eigenvalues can be mitigated [52]. The spectral decomposition of the sample covariance matrix is first formed as:

$$\Sigma = U^{\mathrm{T}} \Lambda U \tag{3.21}$$

Then each eigenvalue is adjusted towards the mean eigenvalue using:

$$\hat{\lambda}_i = \alpha \lambda_i + (1 - \alpha)\frac{1}{k}\sum_{j=1}^{k} \lambda_j \qquad (3.22)$$

where λ_i is the ith eigenvalue from Λ, $\hat{\lambda}_i$ is the adjusted estimate of the ith eigenvalue and k is the number of dimensions. The parameter α defines a family of user models, and the optimal value must be determined experimentally. We describe an experiment to find the value of α which produces the best average precision for a text retrieval test collection in Section 3.3. Once a value for α is chosen, the "regularized" covariance matrix is computed as:

$$\Sigma_\alpha = U \Lambda_\alpha U^{\text{T}} \qquad (3.23)$$

where Λ_α is the diagonal matrix with $\{\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_k\}$ on the main diagonal. The results of an experimental determination of suitable values for α are described in the next section.

Using equations (3.17), (3.18), (3.21), (3.22) and (3.23), the Gaussian User Model is implemented by rank ordering documents in increasing order of the estimated Mahalanobis distance m_α

$$m_\alpha^2 = p_i^{\text{T}} \Sigma_\alpha^{-1} p_i \qquad (3.24)$$

between the projection p_i of any vector v_i and the distribution specified by μ and Σ_α. We describe an experimental procedure to discover a suitable value for α in the next section.

3.3   Experiment Design

The two goals of our experiment were:

- Determine the optimum value for the parameter α in the Gaussian User Model.

- Compare the effectiveness of the Gaussian User Model with the effectiveness of the LSI-mean technique.

It turns out that the same experiment design actually suffices for both purposes. Our basic approach is to repeat the same text filtering experiment, using different values for α each time, and then select the value of α which maximizes the average precision. When $\alpha = 1.0$, the Gaussian User Model produces the same rank order as the cosine measure (a fact which we prove in the next section), so by including 1.0 in the range of values for α both objectives can be realized.

In our experiment we have used the Gaussian User Model to select abstracts from the Cranfield collection. The Cranfield collection contains 1398 abstracts of journal articles from the field of aerospace engineering.[14] Originally designed for evaluation of text retrieval systems, the collection also includes 225 brief topic specifications and binary relevance judgements for every document against each topic. There are between 2 and 39 relevant documents for each topic, with an average of approximately 8.

The Cranfield collection is an attractive choice for this experiment because it is small enough to permit experiments to be run repeatedly and yet there are enough topics so that performance differences on individual topics can be smoothed out in order to understand the "typical" performance of a filtering technique. Given the relatively small number of relevant documents for each query, it is only possible to evaluate the initial learning behavior of the Gaussian

---

[14]The Cranfield collection is available from ftp://cs.cornell.edu/pub/smart/cran

User Model using the Cranfield collection. But because the performance of a filtering technique during the initial training phase is of particular significance in interactive applications, we are willing to accept this limitation.

Hull has used this collection for text filtering experiments as well [67]. For these experiments Hull developed a cross-validation technique to maximize the size of each training set, and we have applied the same technique. For each topic, we repeatedly train on all but one of the relevant documents, and then determine the rank that would be assigned to the remaining relevant document. This process is repeated as many times as there are documents relevant to a topic, each time with a different relevant document withheld from the training set. The result is a list of ranks at which the relevant documents would have been placed if that document had been the last to arrive. In the event of a tie we arbitrarily increment the rank of all but one of the documents, repeating the process until all ties have been removed. For a topic with 8 relevant documents, this is a fairly good estimate of the ranks that would be assigned to a set of 8 relevant documents which arrive after 7 earlier documents have been used for training. As a measure of effectiveness we compute precision using 11 point interpolation (nonincreasing step function interpolation at values between 0.0 and 1.0 in steps of 0.1), average the results over the 225 topics at each of the eleven points, and then average the eleven resulting values to find the average precision.

We used version 11.0 the SMART text retrieval system to conduct the experiment. SMART is designed specifically to support the evaluation of vector space text retrieval systems. As a result, it contains extensive facilities for lexical analysis, vector construction, term weighting, similarity calculations, ranked

output, and computation of evaluation measures. For the Singular Value Decomposition we integrated the SVDPACK sparse matrix SVD routines. Appendix A describes the modifications which we made to the two packages to integrate them, implement the Gaussian User Model, and collect experimental results for text filtering using cross-validation. The SMART and LSI parameters listed in Table 3.10 were used for every experimental run.

Parameter	Value
Dimensions Retained (k)	100
Stopword List	SMART English
Term Weight Function	SMART ltc weights

Table 3.10: Parameters for the Gaussian User Model experiment.

3.4 Results

Figure 3.7 shows the average precision achieved by the Gaussian User Model for a range of values of α. The upper line shows the comparable results when the "ltc" weighting function is applied (including the length normalization) before the SVD is performed. The lower line shows filtering effectiveness when no weighting function or length normalization is performed before computing the SVD. This corresponds to the use of raw term frequency. Dumais has reported that advanced weighting functions significantly improve the performance of LSI for information retrieval, and these results confirm that information filtering applications of LSI accrue similar benefits [42].

Figure 3.7 convincingly demonstrates that the optimum value for α is 1.0, regardless of whether "ltc" or "nnn" weights are used. That is the value, however,
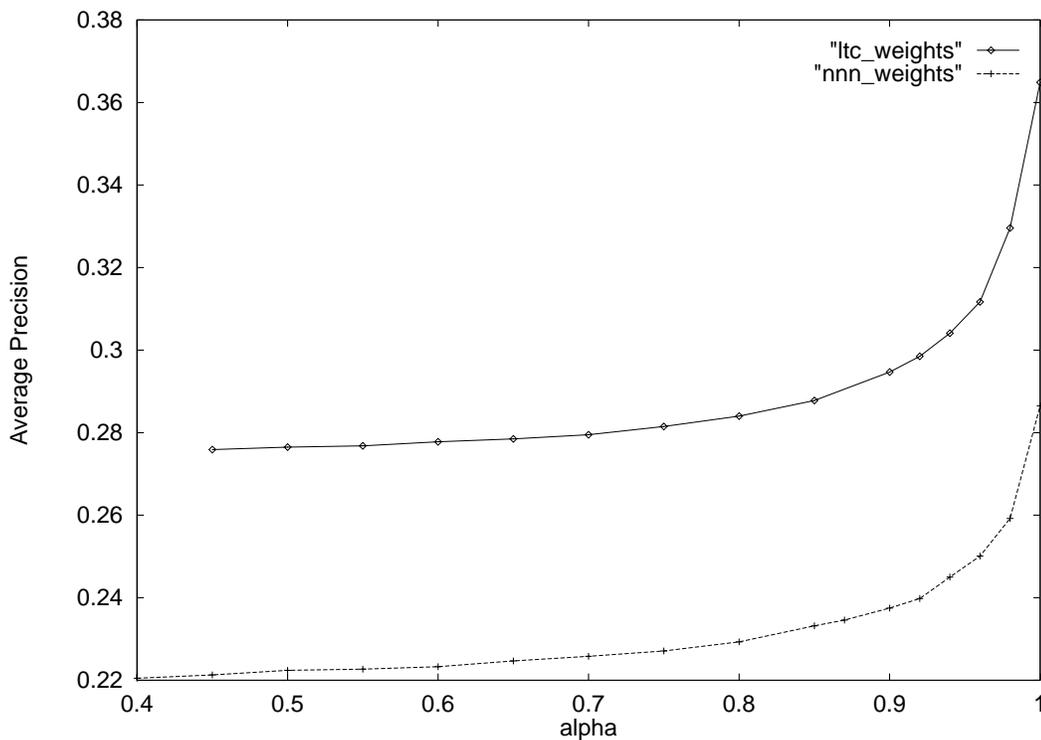
Figure 3.7: Average precision on the Cranfield collection with and without term weights.

which produces the same rank order as Dumais' LSI-mean technique. When $\alpha = 1.0$, a scaled identity matrix is used in place of the covariance matrix, so the Mahalanobis distance between the distribution and any projected vector is a constant multiple of the length of that projection. As Figure 3.8 illustrates, the length $m_{1.0}$ of the projection p_i of any vector v_i will have a length specified by the tangent of the included angle θ (since μ has unit length), provided only that v_i forms an acute angle with μ. Therefore,

$$m_{1.0}^{(1)} > m_{1.0}^{(2)} \Leftrightarrow \quad \theta^{(1)} > \theta^{(2)} \quad \Leftrightarrow \cos(\theta^{(1)}) < \cos(\theta^{(2)}) \qquad (3.25)$$

$$m_{1.0}^{(1)} = m_{1.0}^{(2)} \Leftrightarrow \quad \theta^{(1)} = \theta^{(2)} \quad \Leftrightarrow \cos(\theta^{(1)}) = \cos(\theta^{(2)}) \qquad (3.26)$$

$$m_{1.0}^{(1)} < m_{1.0}^{(2)} \Leftrightarrow \quad \theta^{(1)} < \theta^{(2)} \quad \Leftrightarrow \cos(\theta^{(1)}) > \cos(\theta^{(2)}) \qquad (3.27)$$

since $\tan \theta$ is strictly increasing and $\cos \theta$ is strictly decreasing in the first quadrant. So the order produced when documents are ranked by increasing Euclidian distance in the hyperplane normal to μ will be identical to the order produced when documents are ranked by decreasing values of the cosine similarity measure.
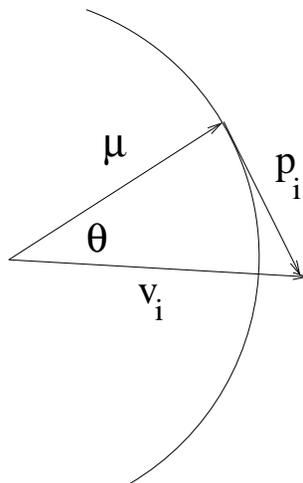
Figure 3.8: Relation of the length of p_i to the included angle θ.

So the somewhat surprising result of our experiment is that the best value of α is the one which reduces the Gaussian User Model to the LSI-mean technique. The effect is even more dramatic near the top of the ranked list (i.e., at lower values of recall), which is the region most likely to be of interest for interactive applications. Figure 3.9 shows the effect of *alpha* on the precision at several levels of recall.

Because the bias towards extreme eigenvalues is most severe when there are few samples, it would be reasonable to consider making α a function of the number of samples in the training set. Unfortunately, no such trend is seen when performance statistics are collected with the topics grouped by the number of training examples. In fact, the same dramatic improvement near $\alpha = 1.0$ is seen
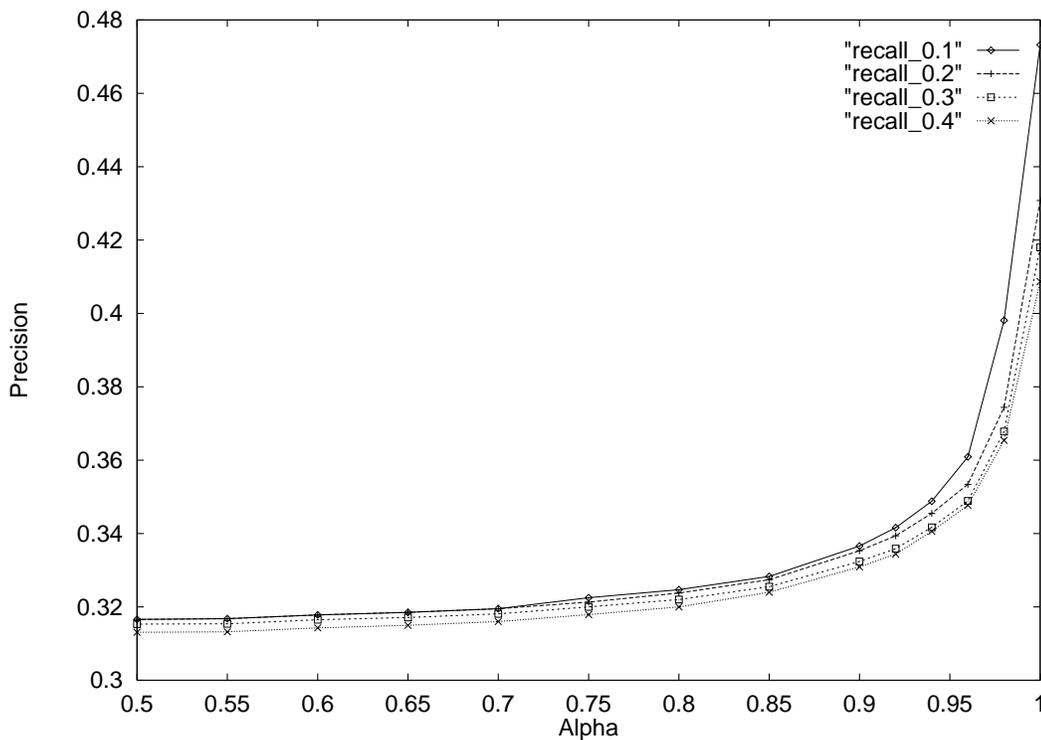
Figure 3.9: Precision at several values of recall, averaged over 225 topics.

for almost every topic when the same statistics are collected on a topic-by-topic basis. So the averaging effect of the average precision statistic is not responsible for these disappointing results.

The sharp rise near $\alpha = 1.0$ suggests that there might be some counterintuitive effect which would further improve effectiveness for values of α above 1.0, a process which (by analogy to overrelaxation) we could refer to as "overregularization." Since it makes little sense to construct an estimate of the covariance matrix which is not positive definite, we have chosen to explore the the region in which every eigenvalue remains positive. This condition is assured for every value of α that transforms the largest eigenvalue to a positive value, so we define

α_{max} as:

$$\alpha_{\mathrm{max}} = \frac{\lambda_{\mathrm{max}}}{\lambda_{\mathrm{max}} - \frac{1}{k} \sum_{j=1}^{k} \lambda_j} \qquad (3.28)$$

where $\lambda_{\mathrm{max}} = \max_j \lambda_j$. The value of α_{max} will vary from topic to topic, so we normalize the region $1.0 < \alpha < \alpha_{\mathrm{max}}$, to cover the range $1.0 < \alpha' < 2.0$ as follows:

$$\alpha' = \alpha, 0 < \alpha \leq 1.0 \qquad (3.29)$$

$$\alpha' = 1 + \frac{\alpha - 1}{\alpha_{\mathrm{max}}}, 1.0 < \alpha < \alpha_{\mathrm{max}} \qquad (3.30)$$

Figure 3.10 shows the result of extending the data for the "ltc" weights from Figure 3.7 into the overregularization region. The sharp falloff at $\alpha = 1.0$ clearly indicates that overregularization is not helpful either.
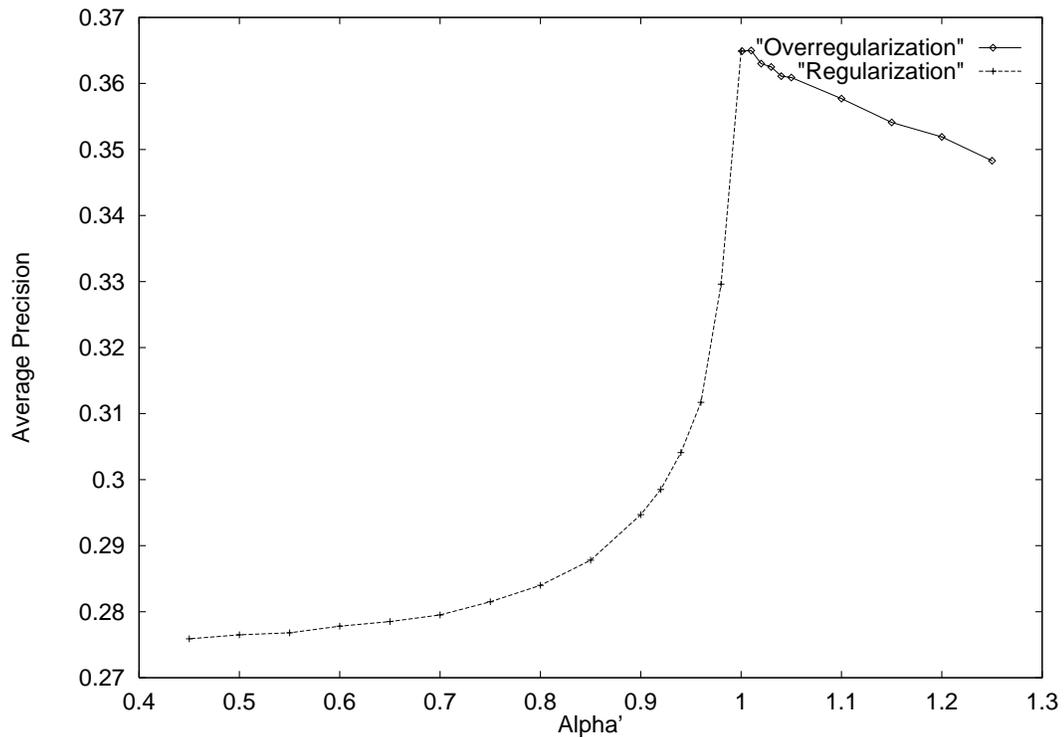
Figure 3.10: Average precision for "ltc" term weights with overregularization.

3.5　Representing Uninteresting Documents

Both the Gaussian User Model and the LSI-mean technique use only positive training examples, those which the user has designated as interesting. From the performance of the LSI-mean technique we can be certain that there is a tendency for similar documents to cluster together in the LSI concept space and that the clustering tendency is sufficiently strong (at least when averaged over many topics) for construction of text filtering systems which exploit it. The Gaussian User Model's failure to improve on (or even equal, except in the degenerate case) the effectiveness of the LSI-mean model suggests that the shape of the distribution cannot be exploited, at least for training sets of moderate size. There are two possible explanations that would be consistent with our hypothesis that human interests are more specific with respect to some concepts than others:

- The LSI concept-vector representation does not preserve sufficient information to recognize the most important concepts with a moderate number of training samples.

- The assumption implicit in the Gaussian User Model that small variance in the training set indicates specificity of interest is invalid.

If the first possible explanation is correct, then no replacement for the symmetric cosine measure can be found and this line of investigation will prove fruitless. The second possibility offers greater promise, since it would be possible to relax the constraint that only positive training documents be used in order to exploit an additional source of information about how concept specificity varies across concepts.

Consider again the original example of documents about emeralds. If the concepts "shiny" and "boat" also appeared in the document collection, they too might be represented in the LSI concept space. "Shiny" might occur often with "emeralds," making it useful for identifying that interest. "Boat," on the other hand, might appear equally rarely in documents about emeralds and those that are not about emeralds. Thus, "shiny" is informative, but "boat" is not. But if both have small variance in the relevant training documents, the Gaussian User Model would treat both identically. The key to recognizing the difference is to represent the expected variations in the non-relevant documents as well and then compare the two interest representations in some way. If that results in improved performance, then we would know that the reason the Gaussian User Model is unable to distinguish between informative and uninformative concepts is that only positive examples are examined.

Unfortunately, the non-relevant documents are unlikely to form a single cluster, so some representation that is appropriate for multimodal distributions must be devised. One approach would be to use the unimodal distribution which represents the relevant documents to define a "region" of interest, and then remove the elements of the multimodal distribution of non-relevant documents which lie outside that region. If the remaining nonrelevant distributions then form a single cluster, discriminant analysis techniques that are appropriate for separating unimodal distributions can then be applied. Hull has developed a technique called "local LSI" that works in exactly this way [68].

Hull defined the "region" of interest on the surface of the unit hypersphere by using the basic (not LSI) vector space method, combining the vector representing the topic specification with the vectors representing relevant documents to form

the profile for the first pass. The top 2,000 documents were then selected (using the cosine measure and that profile), and performed the SVD for LSI using only those documents. The advantage of this preprocessing step is that the resulting LSI representation is tuned to preserve significant differences among the documents which are similar to first-pass profile.

Hull then represented the relevant and non-relevant document sets separately, finding separate mean vectors and covariance matrices, and then computed the Mahalanobis distance from every vector to each of the distributions. These values were used to rank documents in order of decreasing difference between two Mahalanobis distances, with the documents closer to the relevant distribution being assigned positive values so that they would be listed first. This is a variation on Quadratic Discriminant Analysis.[15]

Hull computed mean vectors and covariance matrices in the full LSI space, rather than in the projection to the tangent hyperplane. When some dimensions exhibit large variance, this results in a component of the covariance matrix that is aligned with the mean vector, favoring documents which are closer to the mean by predicting a somewhat larger variance for them. Since this tendency towards the mean is the same effect achieved by increasing α in our model, the two approaches should produce results which are comparable even though the associated value of α may differ.

Hull found no value for α which resulted in a statistically significant improvement in performance over the LSI-mean technique. He ran those experiments by training on 100,000 Wall Street Journal articles which had been judged against

---

[15]In Quadratic Discriminant Analysis a threshold of zero is applied to the difference in order to separate the two sets without computing a rank order.

40 topics for the Text Retrieval Conference (TREC) and then using another 80,000 articles from the same collection for evaluation [68]. This would suggest that on larger collections even the non-relevant documents in the local region are not well represented by a unimodal distribution.[16]

In the more recent experiments that we referred to at the outset of this chapter, however, Schütze, Hull and Pedersen have reported significant effectiveness improvements, even when a significantly simpler technique based Linear Discriminant Analysis is used. In Linear Discriminant Analysis the covariance matrix of the two distributions is assumed to be identical, so every sample (relevant and non-relevant) is used to estimate a single covariance matrix. The mean vectors are, of course, allowed to be different so that the two distributions can be separated.

The key to achieving the effectiveness improvements they report is that only the 250-word passage of each document that is most closely related to an explicit topic description is used to form the term-document matrix when computing the local LSI representation. It would appear from their results that shorter documents provide a more informative basis for recognizing similarities in the usage of individual terms. Such an observation is consistent with the excellent results which Hull obtained when Quadratic Discriminant Analysis was applied to the Cranfield collection, since the median document length in that collection is about 100 words [67].

---

[16]Hull actually did demonstrate improved performance on the Cranfield collection with a similar technique [67]. Hull explains this seeming contradiction by observing that it is far easier to separate the two distributions on the much smaller Cranfield collection.

3.6 Implications for Future Research

The results obtained by Schütze, *et al.* suggest that improved representations, such as those based on identification of salient passages, can produce enough information to permit discriminant analysis techniques which assume unimodal distributions to succeed. Of course, such an approach requires that both relevant and non-relevant training examples be available. The difficulty of collecting explicit judgements about non-relevant documents leads us to conclude that further research on implicit feedback techniques for interactive text filtering will be particularly important if we are to eventually achieve the best possible effectiveness from vector space text filtering systems.

There is no *a priori* reason to believe, however, that the unimodal statistical representations used in Linear and Quadratic Discriminant Analysis capture all of the useful information well, even when the local LSI approach is adopted. In order to investigate whether even better effectiveness can be achieved, it would be useful to experiment with non-statistical techniques based on multimodal representations. DeClaris and his colleagues have developed a heuristic approach which uses successive feature extraction steps (or a neural network) to identify hyperellipsoid or hypercube representations that satisfy logical constraints on cluster formation that are known *a priori* and they have used the technique to construct cognitive models for disease classification and differential diagnosis [22, 35, 33, 61, 70, 146, 151]. In machine learning this type of approach is referred to as "disjunctive induction," since (in text filtering applications) the profile would be built by applying inductive learning to infer a union (or "disjunction") of unimodal representations [82]. Recently developed "neuro-fuzzy" techniques which exploit ideas from both neural networks and fuzzy logic may also be well

suited for adaptive text filtering [36]. Radial basis function neural networks and wavelet neural networks offer other alternatives. All of these techniques need to be investigated in order to discover the ultimate limits on the effectiveness of vector space text filtering.

A more practical question which is raised by our Gaussian User Model experiment is whether improved performance would be observed if more relevant training examples were available. This is the basis for our participation in the "routing" portion of the TREC-5 evaluation, the results of which will be available in November of 1996. If that experiment reveals that a value of α below 1.0 produces good results on large training sets then a hybrid approach in which the LSI mean is used initially, transitioning to the Gaussian User Model once its lower efficiency is offset by improved effectiveness.

Another important issue raised by our work with the LSI-mean technique and the Gaussian User Model is whether the Singular Value Decomposition can be replaced by an alternative technique for generating short feature vectors that can serve as a useful basis for adaptive text filtering applications. Berry has achieved significant savings in computational complexity when a recursive approximation technique is applied by using what are known as "ULV" and "URV" decompositions [8]. We have evaluated an alternative approach based on a heuristic approximation to multidimensional scaling known as "FastMap" that was developed by Faloutsos and Lin [45]. Although Faloutsos and Lin reported promising results on very small document collections, our initial experimental results on the considerably larger Cranfield collection are disappointing [101].

3.7 Summary

We have shown that Dumais' LSI-mean technique is better than any other known vector space approach that uses LSI feature vectors for adaptive text filtering when only a small number of explicitly identified positive training examples are available. LSI feature vectors are important for our adaptive multilingual text filtering experiments and restriction to a small number of positive training examples closely matches our conception of a typical interactive text filtering task, so we can be confident that the LSI-mean technique is a reasonable basis for the experiments reported in Chapter 5. We are now prepared to justify our claim that LSI provides a useful basis for multilingual text filtering, and to begin our search for other promising techniques. This is the subject of Chapter 4, in which we step back and take a comprehensive look at techniques which have been developed for multilingual text retrieval.

Chapter 4

Multilingual Text Retrieval

In this chapter we review the present state of the art in multilingual text retrieval (a topic we have previously surveyed in [104]. Our objective here is to identify techniques that can be adapted to perform multilingual text filtering. By "multilingual" text retrieval we mean the retrieval of documents based on explicit queries formulated by a human using natural language, regardless of the language in which the documents and the query are expressed. Neville has called this a "multilingually searchable system" [98]. We emphasize here only the cross-language aspect of multilingual text retrieval, the case in which queries are expressed in a language different from that of the documents, since our ultimate goal is to craft techniques which can be adapted to select documents in one language based on profiles constructed using relevant training documents that may have been written in another language.

Multilingual text retrieval has been the subject of a good deal of study because there are important information needs which cannot be satisfied by monolingual text retrieval systems. The examples which follow are meant to be illustrative, rather than exhaustive, but together they provide some insight into the

practical problems which have motivated this research.

- A collection contains documents in such a large number of languages that it would be impractical to form a query in each language.

- The documents themselves are expressed in more than one language. Consider, for example:

  - Technical documents in which English jargon appears intermixed with narrative text in another language.

  - Literary criticism which quotes substantial portions of a work in a different language.

  - Academic works which cite the titles of documents in different languages.

- The user is not sufficiently fluent in a document collection's language to express a query in that language, but is able to make use of the documents that are identified. This would certainly be useful for a user who is able to read but not to write well in the document collection's language, but there are a wide variety of circumstances in which a reader totally unfamiliar with the principal language of the document collection might find multilingual retrieval useful. For example:

  - A collection of images that are indexed by captions in a language that is unfamiliar to the user.

  - A researcher seeking to determine which individuals and institutions have conducted research on a particular topic.

– A user with sufficient resources to translate the selected documents into a language that he or she is able to understand.

This last example points up a synergistic relationship between machine assisted translation and multilingual text retrieval. Multilingual text retrieval can be used to reduce the number of documents requiring translation, while machine assisted translation makes it practical to translate the selected documents at a reasonable cost. Incremental improvement in either technology should result in a greater demand for both. A similar relationship exists between multilingual text retrieval and fully automatic machine translation. Although (except in narrow domains such as weather reporting) translations produced by fully automatic systems are of significantly lower quality than machine assisted translations, they can be used in a "screening" role during document selection [97].

Figure 4.1 illustrates how fully automatic and machine assisted translation resources could be integrated with a multilingual text retrieval system. With such a system, queries can be constructed in whatever language the user finds convenient, and documents will be returned in whatever language they are expressed. If necessary, fully automatic machine translation can be used to produce screening-quality translations that allow the user to select documents. When a higher quality translation is required, selected documents can be submitted for machine assisted human translation.

Before proceeding it might be useful to identify related research that is outside the scope of this survey. The term "multilingual" is also commonly used to refer to text retrieval systems which can be parameterized to search in one of several languages (c.f. [26]). In such systems both the query and the documents must be expressed in the same language, so such systems are actually
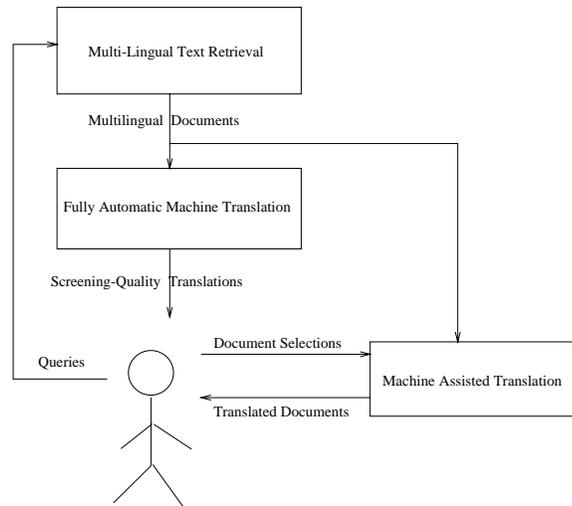
Figure 4.1: Integrating multilingual text retrieval with machine translation.

monolingual text retrieval systems. It is possible to use several monolingual text retrieval systems to retrieve documents from a multilingual document collection, but we do not consider such an approach multilingual text retrieval in the sense of our original definition.

Occasionally, "multilingual" is used even more broadly to describe features of the user interface that allow text to be entered and/or displayed using more than one language or character set (c.f. [116]). This concept is also referred to as "localization" or "internationalization" of software, reflecting the motivation behind the design of a linguistically parameterized user interface. In this context, an online library catalog might be described as "multilingual" if it allowed the user to select the language in which help screens are displayed, even if only monolingual searching is possible.

These closely related research areas offer important perspectives on text retrieval in languages other than English that would be useful to developers of truly multilingual text filtering retrieval systems. Many components of a multilingual

text retrieval system, such as character coding, font construction, morphology, and phrase recognition, can be initially investigated in the context of monolingual text retrieval and then later applied to multilingual text retrieval. But our interest is in cross-language text retrieval. So in this survey we restrict our attention to techniques for selecting documents in one language based on queries expressed in another, and we subsequently use the term "multilingual text retrieval" to mean exactly that.

4.1   Text Retrieval System Model

Figure 4.2 shows a text retrieval system model which is nearly identical to the text filtering system model in Figure 2.3. We have simplified the presentation slightly by restricting that the range of both the comparison function c and the human judgement function j to $[0, 1]$, the set of real numbers between zero and one, rather than $[0, 1]^n$ because none of the multilingual text retrieval systems we describe base their output on more than a single value. But the only significant difference is that in the filtering model the system seeks to develop a profile which represents what Taylor has called the "visceral" information need, while in the retrieval model the queries (Taylor's "compromised" information need [149]) can be obtained directly from the user. We emphasize that in the text retrieval system model by referring to the "query space" Q rather than the "information need space" I, and replacing the "profile acquisition" function p with a simpler "query representation" function q that typically functions in a manner fairly similar to that of the document representation function d.[1]

---

[1] The document representation function's effect is often referred to as an "indexing" because the results of applying d to each document in the collection are often used to construct an
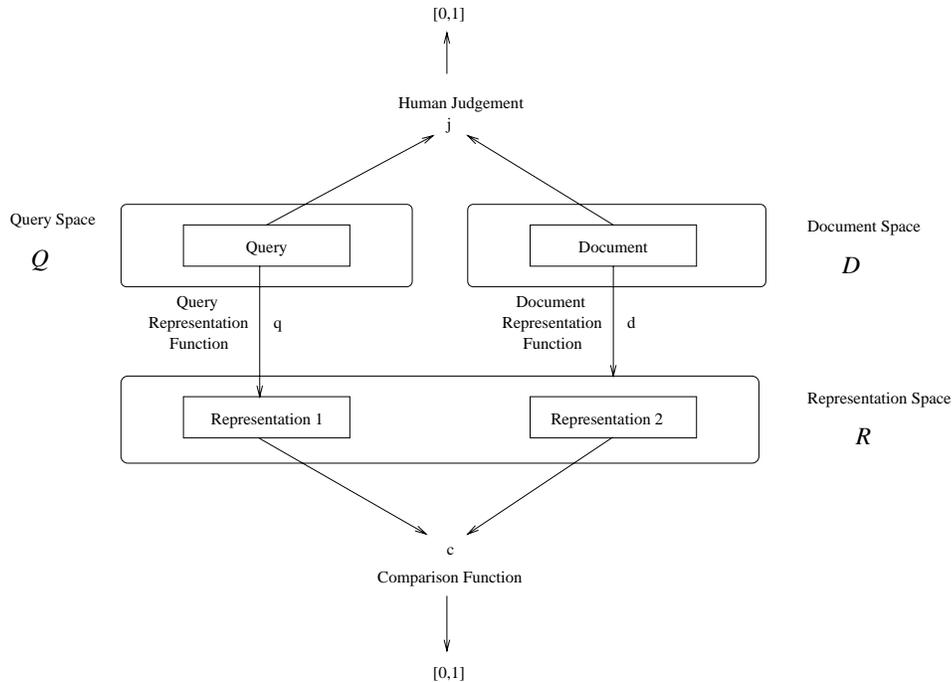
Figure 4.2: Text retrieval system model.

The functions q and d are usually not identical, however, because it is usually helpful to account for observed differences in the characteristics of queries and documents. For example, queries are often quite short (with lengths of one or two words not being uncommon), while documents might easily be hundreds of pages long. Another issue is that users frequently adopt a vocabulary that differs significantly from that in the documents that contain the information they seek [53]. This is known as the "paraphrase problem." One way that text retrieval systems accommodate such differences is by constructing representation functions that treat queries and documents differently to arrive at compatible representations. This distinction is, in fact, crucial for multilingual text retrieval, because the choice of an entirely different language is actually simply an extreme

index of some sort to improve query-time efficiency.

case of this paraphrase problem [46].

4.2 Approaches to Multilingual Text Retrieval

We next present a taxonomy of multilingual text retrieval approaches. Our survey in this case is considerably more comprehensive than our review of adaptive text filtering systems in Chapter 2 In that chapter we sought to identify the important issues and principal approaches for adaptive text filtering so that we could choose a single technique for further study. In this case our goal is to identify every multilingual text retrieval technique that could be adapted to multilingual text filtering in order to choose a representative set of techniques on which to base our experiments. The other significant difference between the two chapters is that average precision measurements have been reported for many more of the techniques we survey here. This should not be too surprising, since evaluation of text retrieval systems is somewhat more straightforward than text filtering system evaluation because retrieval effectiveness on a fixed size collection is more representative of typical applications. Care should be taken when comparing reported average precision values, however, since there are several techniques by which they can be computed. For example, averages over five (0.1, 0.3, 0.5, 0.7, 0.9), nine and eleven recall points are reported by different systems and (because there may be fewer relevant documents than recall points) a monotonically decreasing step function is often (but not always) used to calculate the precision at the chosen values of recall. Where it is necessary to specify such details, we often do so in a footnote in order to avoid burdening our descriptions with excessive detail.

We have identified two main themes in the research literature on multilingual text filtering: knowledge-based approaches and corpus-based approaches. We begin by describing a knowledge-based approach that we call "text translation," both because it is straightforward and because it has a clear potential for application to multilingual text filtering.

4.2.1   Text Translation

Perhaps the most straightforward approach to multilingual text retrieval is to implement either the query representation function q or the document representation function d using a fully automatic machine translation system in order to bring the query and the document into a representation space R that is based on a single language. Surprisingly, although this approach has been suggested repeatedly in recent years [29, 46, 148] we are aware of only one experiment for which results have been reported [47, 46, 117, 118].

One weakness of present fully automatic machine translation systems is that they are able to produce high quality translations only in limited domains. Fluhr observes that text retrieval systems are typically more tolerant of syntactic than semantic translation errors, but that semantic accuracy suffers when insufficient knowledge about the way in which words are used in the appropriate knowledge domain is encoded into a translation system [46]. Since encoding domain knowledge can be expensive, Fluhr's observation would suggest that the effectiveness of a machine translation approach to multilingual text retrieval will be limited, particularly when it is the relatively short queries that are translated.

It might be possible to partially mitigate this problem by translating the documents rather than the queries. Because the documents are typically much

longer than queries, a machine translation system embedded in d would have considerably more contextual information on which to base semantic choices than one embedded in q. Furthermore, text retrieval systems are typically tolerant of occasional semantic inaccuracy if the dominant pattern of the semantic choices is appropriate. Longer documents usually include a larger vocabulary, and a large vocabulary could improve the prospects for developing a dominant pattern of correct semantic choices.

However, the efficiency of available machine translation becomes an issue when a translation system is embedded in d, because d typically must be applied to a very large number of documents. Moreover, some of the work done by a machine translation system yields no improvement in retrieval effectiveness. For example, translation of text requires choosing word order and adding closed class words in the target language.[2] But both of these features are typically removed by q and d.

In fact, some of the work done by a machine translation system could actually reduce some measures of retrieval effectiveness. Because word senses may not be grouped with words in the same way in different languages, machine translation systems attempt to make the best possible determination of the sense in which polysemous words are used.[3] Following that analysis a single sense is chosen for each polysemous word. In a text retrieval system, however, q and d can be designed to preserve information about uncertainty and c can be designed to exploit that information to improve effectiveness. As a simple example of this,

---

[2]Closed class words, words which carry little content, are typically removed by the "stop-word" list in a text retrieval system.

[3]Polysemous words are words which have more than one meaning.

an exact match text retrieval system could substitute every possible translation for a polysemous word, thus increasing recall (at the expense of precision). Some types of ranked retrieval systems are able to represent and exploit information about the probability that each sense of a polysemous word is correct. If this information could be extracted from the machine translation system, average precision might be improved by increasing recall while limiting the adverse effect on precision.

These observations suggest that when designing q and d functions for multilingual text retrieval, the type and depth of processing should be determined by the ability of the representation space R to represent the results of that processing and the ability of the comparison function c to use that information. We could either constrain our processing by the ability of existing techniques to use the resulting information or we could design new representations and comparison functions to exploit the information that machine translation technology can provide. In the remainder of this section we will describe how these two approaches have been integrated in both practical and experimental systems.

4.2.2 Multilingual Thesauri

In this survey we define a thesaurus broadly as any tool that encodes knowledge about a domain by organizing the way terminology is used by an application. In other words a thesaurus is a type of an ontology, one which is specialized to the organization of terminology. A multilingual thesaurus is one which organizes terminology from more than one language. Bilingual dictionaries, which typically define terms with respect to other terms, are clearly subsumed by this definition. Lexicons in computational linguistics, which encode syntactic and

semantic information about terms, are included as well. Complex thesauri used as a concept index in automatic text retrieval systems, are also within the scope of our definition of a thesaurus. Even a simple bilingual listing of technical terms in which each term is assigned a unique translation, would be a thesaurus by our definition. We realize that this is an unusually broad definition of the term "thesaurus." But because no standard terminology succinctly captures the concept we describe, we have chosen to use the term most closely associated with present multilingual text retrieval practice. Table 4.1 shows some common types of thesauri used in multilingual text retrieval systems.

Thesaurus Type	Characteristics
Subject Thesaurus	Hierarchical and associative relations. Unique term assigned to each node.
Concept List	Term space partitioned into concept classes.
Term List	List of cross-language synonyms.
Lexicon	Machine readable syntax and/or semantics.

Table 4.1: Examples of multilingual thesauri.

Thesaurus-based techniques share certain advantages and limitations. Because thesauri can represent relationships between terms and concepts in a way that humans find understandable, thesaurus-based text retrieval allows users to exploit insight gained during the search process to reformulate better queries. Furthermore, because a significant amount of domain knowledge can be encoded in the thesaurus, in the hands of a skilled user a thesaurus-based text retrieval system can be a powerful tool. On the other hand, use of a thesaurus imposes an *a priori* limitation on both the vocabulary the user may employ and on the do-

main to which the text retrieval system can be applied.[4] Present techniques for thesaurus construction and maintenance are resource-intensive, and the training and effort required to effectively use the concept relationships contained in a sophisticated thesaurus can be substantial. We discuss some of these limitations in more detail at the end of section 4.2.2 after we have described how thesauri are used for multilingual text retrieval.

Several aspects of domain knowledge can be encoded in a thesaurus. The key feature of every multilingual thesaurus is a specification of cross-linguistic synonymy.[5] Hierarchical concept relationships (broader term, narrower term) and associative relationships (related term, synonymous term) are typically included in more sophisticated thesauri.[6]

Thesauri can be used either manually or automatically. In so-called "controlled vocabulary" systems, every concept is labeled with a unique descriptive term so that the user can manually specify the appropriate concepts in his or her query. When the concept relationships encoded in a thesaurus are used automatically, the technique is often referred to as "concept retrieval." In a simple concept retrieval system a concept list could be used to replace each term with its concept class to increase recall (again at the expense of precision). A more sophisticated approach, known as "query expansion" would be to use the con-

---

[4]Even fairly comprehensive dictionaries lack detailed coverage of a large number of domains, an observation confirmed by the development of countless specialized technical dictionaries.

[5]The specification of cross-linguistic synonymy need not be complete because some terms may not have direct translations in another language.

[6]Systems which do not make the thesaurus accessible to the user may use only an internal representation for nodes in a conceptual hierarchy, so the "broader terms" we refer to may not be intended for human use.

cept relationships encoded in the thesaurus to choose terms that could improve both precision and recall. We give examples of both techniques below.

Both concept substitution and query expansion represent attempts to increase recall by reducing the effects of the paraphrase problem. Precision can be increased by including syntactic or semantic information in a thesaurus to mitigate the effects of polysemy.[7] For example, in a controlled vocabulary system semantic information (called a "scope note") is often provided in the thesaurus to help users manually choose the correct term. A concept retrieval system could apply this idea by automatically tagging some words with their part-of-speech and then select translations that are appropriate for that part-of-speech. We describe such a system below.

We begin our discussion of thesaurus-based systems with a description of two important early experiments that demonstrated the potential of that approach. We will then describe developments in controlled vocabulary and concept retrieval systems, followed by a description of projects which have exploited encoded semantic knowledge.

Early Work

Pigur describes a multilingual controlled vocabulary thesaurus in English, French and German that was developed for the International Road Research Documentation (IRRD) system in 1964 [112]. But the earliest reported experimental results on the effectiveness of multilingual text retrieval were reported by Salton at Cornell University in 1969 [128]. Salton augmented the SMART text re-

---

[7]Polysemy is the assignment of more than one meaning to a single term. Polysemy resolution is often referred to as "word sense disambiguation."

trieval system with a multilingual concept list constructed by translating some of the words in an existing English concept list into German. Forty-eight English queries for a collection of library science abstracts were manually translated into German, and all four possible language pairs were evaluated. On the 468 German abstracts, the use of English rather than German queries reduced the average precision[8] from 0.35 to 0.34 (3%),[9] while on 1095 English abstracts the use of German rather than English queries reduced the average precision from 0.33 to 0.31 (6%). From this Salton concluded that although retrieval effectiveness varied across document collections (a well known phenomenon in text retrieval), "cross-language processing ... is nearly as effective as processing within a single language." After examining the retrieval failures in more detail Salton concluded that "it would therefore seem essential that a more complete thesaurus be used under operational conditions for future experiments."

For a 1973 paper Salton implemented an English-French multilingual concept list, this time achieving more complete coverage by independently developing the section for each language after establishing a common set of concepts [127]. Again, no information about the relationships between concepts was encoded or used. In this study Salton obtained a French-English parallel corpus of 52 abstracts about documentation and used a set of 16 translated queries.[10] Salton

---

[8]In these studies Salton reported precision at five values of recall evenly spaced between 0.1 and 0.9.

[9]We report average precision to two decimal places, but do not mean to imply that the results are statistically significant to two figures. We report the percentage difference based on these values with reference to the monolingual technique in an attempt to facilitate comparison with other approaches.

[10]A parallel corpus is a collection of documents in which every document is translated into

observed that on French abstracts the use of English rather than French queries increased the average precision from 0.43 to 0.45 (5%) but that on English documents the use of French rather than English queries decreased the average precision from 0.43 to 0.38 (12%). This last result is perhaps explained by the sensitivity of the average precision metric to the rank assigned to a single abstract in such a small collection (a speculation reinforced by the nearly step-function shape of the precision-recall graphs in this case). Salton observes, however, that the smaller English vocabulary in this domain also gave English queries the advantage of operating at a somewhat higher level of abstraction.

At about the same time, Pevzner performed a similar experiment using the Russian PNP-2 exact match controlled vocabulary text retrieval system [111].[11] Pevzner expanded the PNP-2's sophisticated Russian thesaurus, which contained several thousand words, several thousand concepts, and over 600 relationships between those concepts, to English [110]. PNP-2 was then used to retrieve both Russian and English documents based on an identical set of 103 short Russian queries.[12] Using quantities called "losses" and "noise," Pevzner reported that a sign test revealed no statistically significant difference (to 95% confidence) between selections from 4000 Russian and 4400 English electrical engineering documents.[13]

every language.

[11]PNP-2 stands for "Pusto-Nepusto-2." In a 1973 paper [127], Salton translates the name of Pevzner's system as "Empty-Nonempty 2" and transliterates Pevzner's name as "Pevsner."

[12]The examples Pevzner provides are all between 2 and 5 words.

[13]Unfortunately, the cited definitions of "losses" and "noise" are in Russian, and Pevzner's summary of their definition appears to be incomplete.

Controlled Vocabulary Systems

By 1973 it was well established that both controlled vocabulary and concept retrieval systems with multilingual thesauri could achieve performance across languages on a par with the within-language performance of the same techniques. Commercial acceptance soon followed, and by 1977 Iljon was able to identify four multilingual text retrieval systems operating in Europe [72]. Since this early work, six principal lines of research on multilingual thesauri have emerged: design standards, development and maintenance tools, special purpose hardware, new language pairs and domains, user interfaces, and user needs assessment.

In 1970 it was already becoming clear that standardization of thesaurus development to prevent "creation of many divergent and incongruent subject indexing vocabularies" would be beneficial, and in 1971 the United Nations Educational Scientific and Cultural Organization (UNESCO) proposed standards for multilingual thesaurus development [154]. In 1973 the International Standards Organization (ISO)[14] took up the matter, and by 1976 the draft specification had been greatly expanded [3]. Approved in 1978 as ISO 5964 and most recently modified in 1985, the standard describes how domain knowledge can be incorporated in multilingual thesauri and identifies alternative techniques for multilingual thesaurus development. In 1982 the Soviet Union adopted a similar standard, GOST 7.24-80 [108].[15]

The European Parliament's EUROVOC is an example of a modern ISO 5964 multilingual thesaurus [50]. First published in 1984, EUROVOC now includes

---

[14]ISO Technical Committee 46, Working Group 5.

[15]BS 6723, DIN 1463 and AFNOR NF Z 47-101 are the national standards for multilingual thesaurus development in the United Kingdom, Germany and France, respectively.

all nine official languages of the European Community, and portions of it have been translated into additional languages (c.f. [24]).[16] Thesaurus design remains expensive, and this fact has limited the domains to which controlled vocabulary retrieval has been applied. But EUROVOC demonstrates that once the basic concept relationships have been defined for a domain, extension of an ISO 5964 multilingual thesaurus to additional languages is quite practical.

As large multilingual thesauri have proliferated, design and maintenance tools have become increasingly important. In 1970, Neville described a procedure for merging thesauri that could be used to merge monolingual thesauri to produce a multilingual thesaurus [99], and in 1975 Neville contrasted this approach with other ways of producing multilingual thesauri [100]. Bollmann and Konrad presented a technique for merging monolingual with bilingual thesauri in 1975 [13], and in 1977 Iljon surveyed available thesaurus design and maintenance tools and described the operation of the Commission for the European Communities' ASTUTE system [71].

More recently, an automatic technique for using a thesaurus to generate corresponding indexing terms in four languages was described by Pelissier, *et al.* in 1986 [109]. In 1987 Kalachkina presented an algorithm for merging thesauri in different languages [75] and in 1989 Loginov described tools developed in the Soviet Union to maintain a Russian-English version of the (monolingual) United States National Library of Medicine's Medical Subject Heading (MeSH) thesaurus [88]. Loginov's paper illustrates a case in which external factors (changes to MeSH) generate the thesaurus maintenance requirements. Sosoaga of SABINI,

---

[16]The nine languages are Danish, Dutch, English, French, German, Greek, Italian, Portuguese, and Spanish.

a Spanish library automation company, also described the design of interactive tools for multilingual thesaurus maintenance [32]. The SABINI system was designed for automation of bibliographic records in an online library catalog. Sosoaga provided no examples of implementations for specific languages, however.

In 1988 Kitano, from NEC's Tokyo Software Engineering Development Laboratory, described the development of a hardware tool designed to support multilingual text retrieval [78]. Kitano implemented a Japanese-English thesaurus using a NEC integrated circuit known as the "Intelligent String Search Processor." At the time, the ISSP thesaurus implementation had not been integrated with a text retrieval system, however, so no experimental results were reported.

The research literature on multilingual text retrieval offers several examples of systems which have implemented new language pairs [2, 19]. and new domains [6, 83, 156]. Because this type of report can describe the effect of previously unseen linguistic phenomena on thesaurus design and other aspects of a text retrieval system (e.g. stemming and compound recognition), case studies can provide useful insights into the complexity of implementing ISO 5964 and similar national standards.

Semturs, of IBM Netherlands' Scientific and Cross Industry Center, provided some insight into the contemporary commercial development of user interfaces for multilingual text retrieval systems in the mid-1970's [133, 134]. Semturs described the capabilities of a commercial product, the STAIRS-TLS exact match text retrieval system, which was able to accommodate queries and documents in German, English and French. STAIRS was originally a monolingual full-text

retrieval system,[17] and STAIRS-TLS added a multilingual thesaurus. It included an interactive interface with thesaurus-based tools to facilitate controlled vocabulary query formulation. Semturs' papers report no performance figures, but they offer some insight into the market demands for multilingual text retrieval.

More recently, a team at the University of Huddersfield Centre for Database Access Research in the United Kingdom led by Pollitt has integrated multilingual thesauri with interactive personal computer technology to address one of the fundamental limitations of controlled vocabulary text retrieval [12, 86, 113, 114]. Experience has shown that although the domain knowledge that can be encoded in a thesaurus permits experienced users to form more precise queries, casual and intermittent users have difficulty exploiting the expressive power of a traditional query interface in exact match retrieval systems. Adapting their Menu-based User Search Engine (MenUSE) to use the European Parliament's multilingual EUROVOC thesaurus, Pollitt's team has developed a query formulation tool which facilitates visual browsing in the user's preferred language. Pollitt's team has also extended the English thesaurus for the INSPEC database to Japanese and integrated it with MenUSE. The cited works do not report experimental results on the utility of the multilingual MenUSE interface, but a monolingual evaluation of MenUSE on the INSPEC database is presented in [139].

Controlled vocabulary text retrieval systems are widely used in libraries, and user needs assessment has received considerable attention from library and information science researchers. Rolling described a user needs assessment conducted for the Council of the European Community in 1974 [126], and the TRANSLIB

---

[17]A full-text retrieval system is one which can index any word appearing in any document, regardless of whether it appears in a thesaurus.

project, a part of the European Commission's I*M-Europe Telematics for Libraries program, provides a recent example of user needs assessment [147]. TRANSLIB's goal is development of a trilingual (Greek, Spanish and English) subject search capability for an online library catalog. Chachra discussed user needs assessment for multilingual online library catalogs in [21] more generally, and provided examples from the VTLS online library catalog system. In addition to monolingual full text searching, VTLS used a multilingual thesaurus to suggest controlled vocabulary search terms in a second language. Rolland-Thomas described a similar feature in the Canadian DOBIS bilingual online library catalog, and discussed the utility of more automatic techniques from a user needs perspective [125].

Pasanen-Tuomainen, of the Helsinki University of Technology, reported results from a usability assessment for a multilingual online library catalog, TEN TTU, that incorporated both multilingual controlled vocabulary and monolingual full text searching. [107][18] Examining 2,620 search commands issued during 655 sessions, Pasanen-Tuomainen found that library staff used the controlled vocabulary in 46 of their 337 search commands (14%), but that other patrons used it for less than 3% of their commands. Of the remaining search commands, 11% contained words found in the thesaurus that could have been mapped across languages had TENTTU been designed to do so. Pasanen-Tuomainen also suggested that limited thesaurus availability and inadequate patron training might have reduced thesaurus utilization.

Multilingual text retrieval systems are widely used today, but nearly every

---

[18]TENTTU used the Universal Decimal Classification (UDC), a greatly expanded version of the Dewey decimal system, as a multilingual subject thesaurus.

commercial system that we are aware of uses an exact match approach.[19] Sophisticated multilingual thesauri have been developed for many domains and many languages, and the procedures for adding new domains and languages are well understood. Thesaurus-based techniques have a number of limitations, however. Dubois has identified three key factors which together motivate the search for other techniques: cost, usability by untrained users, and effectiveness [40].

Thesaurus construction is an expensive activity. But the use of a thesaurus can be even more expensive than its construction because in a controlled vocabulary system every document must be assigned terms that reflect the concepts it contains.[20]. Although automated tools can improve human productivity, as long as human intellectual activity is required to recognize and organize information the costs will remain substantial. In fact, with the sustained dramatic decline of computer hardware costs, human activities such as thesaurus maintenance and controlled vocabulary indexing have come to dominate system costs. This limits both the scalability of existing thesaurus-based systems to accommodate the rapid growth in electronically accessible texts and the generalizability of the technique to applications such as high-volume broad-domain text filtering or the retrieval of documents from personal collections in which the manual construction and use of a thesaurus may be economically impractical.

Another important limitation of controlled vocabulary text retrieval techniques, and one which is shared by full text exact match techniques as well, is that untrained users seem to have difficulty exploiting their capabilities. Sig-

---

[19]The exception is the SPIRIT system developed for EMIR which we discuss below.

[20]Dubois discounted this factor, but that analysis was conducted in the context of abstracting services in which the cost of abstract preparation dominates the processing cost for newly arrived documents

nificant differences between the performance of skilled and untrained users have been observed with their choice of terms, their use of the term relationships that can be encoded in a thesaurus, and their use of operators such as "and," "or" and "not" for query construction. In many cases it has proven more economical to provide trained intermediaries than to provide adequate training to each user. Advanced user interfaces such as Pollitt's MenUSE system offer some potential for mitigating this problem, and expert systems that construct Boolean queries from natural language have been investigated in a monolingual context [93]. The ranked output techniques we described in Chapter 2 represent another approach to solving this problem. Ranked retrieval systems typically accept queries in natural language and allow a (relatively) unconstrained choice of terms. In general, the goal of ranked retrieval is not to replace exact match techniques but rather to augment them with techniques that improve the search effectiveness of untrained users. In multilingual text retrieval, ranked retrieval techniques also allow us to avoid an unsolved problem identified by Chachra [21], who observed that single terms in one language can correspond to complex boolean expressions in another when a controlled vocabulary is not used.

A third reason to investigate alternatives to traditional thesaurus-based techniques is to improve effectiveness. Language use is a creative activity, and new words enter human languages each year. Because thesaurus construction is time-consuming, thesauri in production applications necessarily lag somewhat behind the common use of terminology. Furthermore, there is some evidence that thesaurus designers have more difficulty anticipating which concepts and relationships will be useful to their system's eventual users than a cursory inspection

of the thesaurus would suggest [129].[21] Since corpus-based techniques are based on the observed statistics of term usage, they offer some hope that important aspects of current term usage can be identified and exploited. The potential of corpus-based multilingual text retrieval techniques has yet to be realized in a large-scale experiment, however, so we will begin our discussion of experimental techniques with those which include some form of human-usable thesaurus.

Concept Retrieval

Concept retrieval systems seek to address some of these limitations by exploiting the information encoded in a thesaurus without human intervention. Salton's early experiments provide one example of concept retrieval [128]. An alternative to Salton's choice to populate the representation space R with representations based on concepts is to populate it with representations based on terms, but to use the multilingual thesaurus to guide the term selection process. This is a variation on query expansion, a well studied technique for monolingual text retrieval [18].[22] The basic idea of query expansion is to accommodate term usage variations by augmenting the terms in the query with related terms. But because query expansion typically improves recall at the expense of precision, selection of inappropriate terms could reduce overall performance measures such as average precision. So, in the context of multilingual text retrieval, the goal of query expansion techniques is to accommodate cross-linguistic term usage variation

---

[21]Discussions about the relative effectiveness of controlled vocabulary and statistical text retrieval are often marked by considerable enthusiasm on both sides, however, so it is difficult to find impartial evaluations on this issue.

[22]The unique feature of cross-language query expansion is that the original term is removed from the expanded query unless it carries the same meaning in both languages.

while minimizing the adverse impacts on effectiveness.

Recently, Davis and Dunning of New Mexico State University have evaluated several multilingual text retrieval techniques, one of which is based on query expansion [29]. For the evaluation of Spanish text retrieval at the fourth Text Retrieval Conference (TREC-4) they manually translated 25 Spanish queries into English and then used them to select documents from a collection of 58,000 Spanish articles from the Mexican newspaper "El Norte" using the INQUIRY text retrieval system. For each of these English queries they then automatically formed corresponding Spanish queries by selecting every English translation for each word in the query from a simple bilingual term list.[23] This approach, which they used as a benchmark against which to compare their corpus-based approaches, achieved an average precision of 0.04. Five of the ten participants in the TREC-4 Spanish text retrieval evaluation achieved an average precision exceeding 0.21 on the same collection by using the Spanish queries directly, so Davis and Dunning's results suggest that unconstrained query expansion is of limited value for multilingual text retrieval.[24]

Building on this work, Hull and Grefenstette at Rank-Xerox in France have evaluated the potential of more sophisticated approaches to query expansion [69]. They manually translated 50 short TREC queries[25] into French and created a bilingual term list that contained every possible translation for each French

---

[23]Davis and Dunning used an online version of the Collins English-Spanish dictionary as a bilingual term list.

[24]The best average precision achieved by a monolingual system was 0.49.

[25]Hull and Grefenstette used shortened versions of TREC queries 51-100 which had an average length of seven words.

word.[26] Unconstrained cross-language query expansion was then used to select from approximately 500,000 newspaper articles for which relevance judgements were available using the SMART vector space text retrieval system. They found that adding phrases[27] to the bilingual term list increased their effectiveness measure[28] from 0.27 to 0.36 (33%).[29] Using the original English queries, Hull and Grefenstette achieved an effectiveness measure of 0.39. From this they concluded that inclusion of phrases in a bilingual term list can allow the query expansion technique to perform almost as well across languages as traditional statistical techniques do in a monolingual setting.[30]

The European Multilingual Information Retrieval (EMIR) project, led by Fluhr of the French Institut National des Sciences et Techniques Nucléaires (INSTN), also used a query expansion technique [47, 117, 118, 119, 141]. An ESPRIT II[31] project, EMIR work proceeded between November of 1990 through March of 1994. The goal of EMIR was to extend the SPIRIT text retrieval sys-

---

[26]The bilingual term list was manually constructed using the third edition of the Robert and Collins French-English dictionary.

[27]Only phrases appearing in the same dictionary were added.

[28]Hull and Grefenstette reported precision averaged over fixed size sets containing the top ranked 5, 10, 15, and 20 documents.

[29]These figures were used by Hull and Grefenstette as a benchmark for evaluating automatic techniques for constructing term lists from an online dictionary that was designed originally for human use.

[30]In comparing these results with those of Davis and Dunning it is important to consider that Hull and Grefenstette selected their effectiveness measure with interactive applications in mind.

[31]ESPRIT II was the second phase of the European Commission's information technology research program.

tem (which was originally developed by Fluhr and others) to multiple languages. The initial language pair was English and French, and it was later extended to German. Analit Ltd., a Russian company, is extending SPIRIT to Russian. SPIRIT is a ranked Boolean text retrieval system, in which sets are selected using successively smaller portions of the original query and then ranked for display in order of increasing generality.

For the French/English language pair there were 33,153 mappings fromFrench terms to one or more English terms. Each such mapping had between 1 and 24 possible English terms, and the median number of English terms for a French term was 2. English terms which did not appear in the document collection were then eliminated. On a parallel bilingual corpus from the European Court of Justice, this achieved at least a 40% reduction in the number of target terms for 92.6% of the mappings. More comprehensive performance results are given below.

Encoding Semantic Information

Another aspect of the EMIR project was application of fast but shallow parsing to exploit semantic information that was encoded in the thesaurus [119, 117, 118, 141]. The number of English terms was reduced by labeling each English term with the corresponding part of speech and then only choosing those English terms which were appropriate for the syntactic usage of the French term. The EMIR thesaurus was a bilingual term list in which semantic information, encoded as compounds, was used in place of concept relationships. In EMIR, terms included words, phrases and compounds. Because compounds link key terms together on the basis of their semantic relationship rather than their surface form, compound

formulation is more powerful than simple phrase extraction. Because the order of the components in a compound was sometimes switched in the target language, the term list entries for compounds were constructed to account for transposition when necessary.

The EMIR version of SPIRIT was evaluated on the Cranfield collection of 1398 aeronautical abstracts using 225 queries which had been translated into French by the French Army Documentation Center. English documents were retrieved in response to French queries. For comparison, the French queries were translated back into English using the SYSTRAN fully automatic machine translation system and documents were selected using a monolingual version of the SPIRIT text retrieval system. EMIR increased average precision over the combination of SYSTRAN and SPIRIT from 0.21 to 0.27 (29%), but use of the original English queries with SPIRIT further increased average precision to 0.34 (26%).[32]

Some more exploratory projects with potential multilingual text retrieval applications have also been reported. Rassinoux's recent work on multilingual text retrieval using conceptual graphs offers some insight into how deep semantic processing might be used [122]. The system, known as RECIT, was designed for the sharply limited domain of radiology reports and hospital discharge summaries from the digestive surgery department at a single trilingual (French, English and German) hospital.[33] Rassinoux developed syntactic and semantic analysis routines to produce conceptual graphs in a manually constructed conceptual schemata, but provided no detail on how these conceptual graphs might be

---

[32]These are nine point averages, evenly spaced between 0.1 and 0.9.

[33]RECIT stands for REpreséntation du Contenu Informationnel des Textes médicaux.

matched. The development of techniques for approximate matching of conceptual graphs would be useful in this regard.

Kitano's 1988 paper described a Direct Memory Access Parser (DMAP) implementation, a system called "SMAP," using the same hardware [78]. SMAP was designed to extract concepts from multilingual sentences and use them to fill a case frame.[34] Reported parsing speeds were better than one millisecond per word for sentences of up to 10 words. Kitano did not, however, discuss how the case frames would be designed (except to observe the need for development tools), or how they would be matched.

Other research projects

In addition to the research cited here, we are aware of two other research groups working on thesaurus-based multilingual text retrieval. Because we know of no published research results from these projects, we simply describe their stated objectives briefly.

In December of 1993 a team led by Laus-Maczynska of the French firm Cap Gemini Innovation began work on the CRISTAL project [20]. A part of the I*M-Europe Language Engineering program, CRISTAL was designed to retrieve documents from a French collection using queries in French, English or Italian using the French Dicologique thesaurus. It was scheduled for completion in May of 1996. In the other project Liddy, of Syracuse University and Textwise Inc., began a feasibility study of multilingual text retrieval for the Advanced Research Projects Agency (ARPA) in 1994 [1]. The proposed system, known as CINDOR, was also designed to exploit a multilingual thesaurus for concept retrieval.

---

[34]A multilingual sentence is one in which words from more than one language appear.

4.2.3 Corpus-Based Techniques

The alternative to use of a thesaurus is to directly exploit statistical information about term usage that can be gleaned from parallel corpora. This more direct approach is well suited for integration with text retrieval techniques that are themselves based on the statistics of term usage.

Automatic Thesaurus Construction

In a sense, corpus-based techniques can be viewed as a type of automatic thesaurus construction technique in which information about the relationship between terms is obtained from observed statistics of term usage. The difference is that in this case the "thesaurus" need not be constructed by humans. As with many other multilingual text retrieval techniques, automatic thesaurus construction has a significant research heritage in a monolingual context [27]. A substantial amount of research has appeared on this subject has been reported in the machine translation literature. For the present survey we describe two techniques for automatically constructing multilingual thesauri from a text retrieval perspective.

The first technique, developed by van der Eijk of Digital Equipment Corporation in the Netherlands, was tested on 1,100 noun phrases drawn from a parallel corpus of about 1000 long Dutch and English sentence pairs in a technical document [155].[35] The noun phrases in each sentence pair were identified using a statistical part of speech tagger and a simple parser. Candidate translations for each Dutch noun phrase were constructed by comparing the frequency with

---

[35]The average sentence length was over 24 words. The sentences were aligned using statistical techniques, and 7% of the sentence pairs were later discovered to be incorrectly aligned.

which each English term occurred in the English portion of sentence pairs containing that noun phrase to the frequency with which that English term occurred in the entire collection. An additional feature was incorporated to discourage the choice of noun phrases which occurred at significantly different relative positions in the sentence pairs.

Parameters were found that resulted in identification of the single correct translation 45% of the time, and alternative choices which produced a list of candidate translations containing the correct single translation 66% of the time were also identified. Sentence alignment, part of speech tagging and parsing errors accounted for 85% of the errors, so van der Eijk speculated that selection of the upper bound on the performance of the technique was a correct single translation about 60% of the time or inclusion of the correct translation in a list about 95% of the time. Because of the small size of the parallel corpus it was not possible to determine the performance of the technique when more than one translation of the same term was present in the corpus.[36] The resulting bilingual lexicon was not used for text retrieval, so we are unable to determine what effect the translation errors would have on retrieval effectiveness. Furthermore, we can offer no guidance regarding whether the precision reduction resulting from increasing the number of candidate translations could be offset by the recall increase resulting from a greater likelihood of including the correct translation in the list. We do, however, present experimental results for a similar technique in Chapter 5 that lead us to believe that van der Eijk's technique may have practical application.

Lin and Chen at the University of Arizona have applied a machine learning

---

[36] 71% of the Dutch noun phrases occurred only once in the entire collection.

approach to multilingual thesaurus construction [66]. Extending earlier work on term clustering, they developed a Chinese-English concept list using a collection of 1052 titles from Chinese technical papers, many of which contained a mixture of Chinese and English words. Using synaptic weights based on the pairwise co-occurrence of terms in the same title, they constructed a Hopfield neural network to generate clusters of terms.[37] Their system clustered terms from 68% of the documents into 36 concepts (without overlap), and they report that manual inspection showed that the terms associated with "all concept descriptors appeared to be relevant and precise" and that some clusters contained both Chinese and English terms. Lin and Chen also suggest that the raw term co-occurrence values could be used directly in a manner similar to the "related term" information in a conventional subject thesaurus. They report no experimental retrieval results, however.

Recently, Sheridan and Ballerini of the Swiss Federal Institute of Technology (ETH Zurich) have successfully applied an automatic thesaurus construction technique to a large scale multilingual text retrieval problem [136]. Rather than use a parallel training corpus, they chose a corpus of approximately 180,000 "comparable" news articles split about evenly between German and Italian that had addressed the same topics on the same day. Feature vectors for each term were constructed using a term-document matrix in the manner described in Chapter 3, but without applying Latent Semantic Indexing to reduce the number of features. The similarity between each German term and every Italian term was then computed using the cosine measure and the results used to con-

[37]In Chinese multiple symbols were recognized as phrases, but in English individual words were used.

struct a list of the 25 Italian terms whose usage was most similar to that of each German term. Augmenting the SPIDER text retrieval system with this term list for cross-language query expansion, Sheridan and Ballerini found that the technique reduced average precision from 0.527 to 0.278 (52%) when compared to the monolingual performance of the same retrieval system with queries in Italian when comparable stemming algorithms were used. They also found that relevance feedback using passages marked by an interactive user increased average precision in the cross-language case by 29%, although they reported no comparable result for the monolingual case.

Vector Translation

Automatic thesaurus construction techniques similar to those used by Sheridan and Ballerini could certainly be useful for multilingual text filtering. As our work with the LSI-mean technique and the Gaussian User Model in Chapter 3 illustrates, however, adaptive approaches to text filtering often represent documents and profiles in ways that are not designed for direct interpretation by humans. Thus, limiting the thesaurus to associating single terms from one language with one or more terms in another language may restrict the application of such techniques to the documents (in which the words are available) rather than to the profile. We now turn our attention to corpus-based multilingual text retrieval techniques which produce mappings that are not designed for human use. In particular, we consider statistical multilingual text retrieval techniques in which the goal is to map statistical information about term use between languages. Vector representations have proven useful for this purpose, so we describe in this section techniques which map vectors from one language to another, a process

we call vector translation.

Fluhr describes a particularly simple technique which provides a good starting point for our discussion [46]. Consider a two language case in which we have three subcollections, one in English, one in French and one which is parallel (i.e., every document in the parallel collection appears in paired English and French versions). Each query is first presented to the parallel collection, and the documents in that collection are ranked with respect to the similarity between the query and the version of the documents that are in the query's language. The highest ranking French documents are then concatenated and used as a query on the remaining French documents, a variation on a technique known as relevance feedback. The same is done for the English documents. The three ranked lists are then combined in some manner and presented to the user.[38]

Relevance feedback is a commonly used technique in statistical information retrieval. A normalized *tfidf* vector is, in a sense, a heuristic approximation to the empirical distribution of term importance within a document. Viewed in this light, the normalized inner product is simply the correlation between two documents described by such distributions.[39] Since the quality of an empirical distribution can be improved by adding observations, relevance feedback can be viewed as a heuristic approach to smoothing out the clumpy empirical distributions that are associated with relatively short queries.[40] In other words,

---

[38]We are not aware of experimental results which describe the effectiveness of this technique.

[39]By linearity, the normalized inner product is the inner product of the normalized *tfidf* vectors.

[40]Proving such an claim would require statistical independence of the observations, a condition that is unlikely to be be satisfied. But relevance feedback has been observed to improve effectiveness, so we seek here to explain, not to prove, its effectiveness.

relatively unimportant terms are suppressed and relatively important terms are reinforced.

In their TREC-4 experiment, Davis and Dunning tried three more complex vector translation techniques[29, 30, 31]. Using 80,000 pairs of aligned sentences from a parallel corpus of United Nations documents, they first selected the 8,000 English sentences that were most similar to their English translations of each TREC query. They then used the Spanish versions of those 8,000 sentences to select 100 common Spanish terms associated with each query.[41] Terms were then adaptively deleted from this set using an evolutionary programming strategy, with a goal of finding a Spanish query that could select Spanish sentences in a way similar to the way the English query selected English documents.[42] Details of the technique are presented in [30]. The evolutionary programming step only increased average precision from 0.004 to 0.02,[43] but they observed that additional improvement might be obtained if a parallel training corpus from a domain more closely related to the evaluation domain were available.

Their third technique was based on the same training corpus of aligned sentences. Davis and Dunning chose the 100 terms with the greatest statistical significance[44] from the set of terms appearing in the Spanish sentences that

---

[41]The 100 terms chosen were those were the 501st to the 600th most common terms.

[42]More precisely, a Spanish query was sought which would maximize the unnormalized inner product of two 80,000-element vectors, one formed by computing the cosine similarity between that Spanish query and each Spanish sentence and the other formed by computing the cosine similarity between the fixed English query and each English sentence.

[43]Recall that they achieved an average precision of 0.04 with unconstrained query expansion.

[44]The statistical significance of each term was estimated using a likelihood ratio test, comparing term frequency in the selected set with term frequency in the entire collection.

were aligned with the 100 sentences most similar[45] to each English query. This technique achieved an average precision of 0.02.

Davis and Dunning's final technique was based on direct translation of vectors [29, 43] using a linear operator. They began by forming one matrix from a collection of *tfidf* vectors derived from the English version of the aligned sentences and a second matrix derived from the Spanish versions of the same sentences. They then solved the resulting underdetermined (and potentially inconsistent) set of vector equations to find a linear operator which translated the Spanish matrix into the English one. They then used that operator to translate each English query's *tfidf* vector into a Spanish *tfidf* vector and used the translated vector to rank the Spanish documents. Davis and Dunning achieved an average precision of 0.01 using this technique. They cautioned, however, that their algorithms for computing the linear operator were still quite preliminary, so much better performance might be possible using this technique.

We have developed another vector translation approach based on parallel corpora which have been aligned to the word level [34, 102], and we report the results of our initial experiments with this technique in Chapter 5. Building on term alignment techniques similar to those used by van der Eijk, we construct a bilingual term list in which alternative translations of each term are assigned (unconditioned) probability values based on the observed frequency with which words align. We then use this statistical bilingual lexicon as a linear operator to map query vectors into another language. We do not believe that this technique would be particularly useful for multilingual text retrieval applications in which short queries are common because, as Hull and Grefenstette have observed, the

---

[45]Again, similarity was computed using the cosine measure.

addition of extra terms would likely seriously degrade recall. As we describe in Chapter 5, however, there is some reason to believe that a statistical vector translation technique based on word alignment would be particularly well suited to the adaptive multilingual text filtering problem.

Latent Semantic Indexing

Another statistical technique that has been applied to multilingual text retrieval is Latent Semantic Indexing (LSI) [37]. LSI has been applied to multilingual text retrieval in a similar way to the relevance feedback technique described above [7, 80, 81, 162]. The basic approach is best illustrated by Landauer and Littman [81]. Randomly selecting 900 training paragraphs and 1,582 evaluation paragraphs from the Hansards collection, a parallel corpus of Canadian parliamentary proceedings, they first applied LSI to identify the principal components of the training set. When LSI is applied to a parallel corpus, the matrix decomposition naturally identifies the principal components in the vector space associated with each language and produces a mapping from each to a common representation space with fewer dimensions. They then selected the principal components of the *tfidf* vector for every paragraph in the evaluation set, regardless of language, in this common representation space. Using the English vectors as queries, they found that the top ranked French vector was derived from the translated version of the English paragraph in 92% of the 1,582 cases. Unfortunately, the lack of a bilingual corpus with available relevance judgements precluded a more traditional recall-precision evaluation.

Berry and Young repeated this work using passages from the Bible in English and Greek [7]. They were able to demonstrate that fine-grained training data,

using only the first verse of each passage to identify the principal components, improved retrieval performance over Landauer and Littman's coarse-grained approach. Using 16 short queries, each of which had between two and six relevant passages in a collection of 734 passages which they constructed.[46] Rather than report precision-recall results they observed that the average rank of a relevant document decreased from about sixth to fourth when the same number of training verses were distributed across every passage in the collection rather than clustered in a small group of passages.

In an interesting combination of corpus-based and thesaurus-based techniques similar to that used by Sheridan and Ballerini, Evans and others at Carnegie Mellon University used LSI to suggest terms from a controlled vocabulary of 125 English medical terms based on natural language queries expressed in Spanish [44]. Augmenting definitions found in three English medical thesauri with related words from both English and Spanish, they obtained a training set of 3,084 words.[47] Their report presents two examples in which the most highly ranked terms would be good choices for use in a controlled vocabulary search, but no multilingual retrieval experiments using this data were reported.

4.2.4 Combined Techniques

Kikui, *et al.* at the Nippon Telegraph and Telephone Corporation (NTT) have recently implemented a technique which combines aspects of both the knowledge-based and corpus-based approaches in a system they call TITAN. Their approach is essentially knowledge-based, using a bilingual term list to translate words

---

[46]The queries contained between one and four words.

[47]Evans, et. al., used term definitions from the QMR, PTXT and UMLS META-1 thesauri.

appearing in a collection of Japanese and English text collected by a World Wide Web search engine [77]. Translation can be done in either direction, depending on the language of the query and the text being searched. Statistical techniques are used to determine the language of each document so that terms requiring translation can be identified. When more than one possible translation for a term is found in the bilingual term list, TITAN consults term frequency statistics compiled from a corpus of World Wide Web pages and selects only the terms which appear most frequently in that corpus.[48] Estimating the recall achieved by a World Wide Web search is a daunting task, so Kikui, *et al.* report no experimental results on retrieval performance. They do report that TITAN received over 15,000 queries per day in May of 1996, however, reflecting an impressive level of market acceptance for a new system.

4.3   Some Observations on the State of the Art

We can take advantage of this extensive background to make a few observations on the present state of multilingual text retrieval practice and research, many of which are equally applicable to multilingual text filtering.

Controlled vocabulary techniques are extremely well developed, but fully automatic thesaurus construction is still in its infancy. Furthermore, multilingual concept retrieval techniques such as query expansion that could exploit information encoded in a thesaurus without human intervention at indexing or retrieval time have thus far been limited to approximating the within-language effectiveness of the same technique in the same domain. Without effective automatic

---

[48]The number of terms that are selected is not specified in their paper.

thesaurus construction, the limited domain of concept retrieval techniques will remain a serious limitation.

The relative immaturity of corpus-based techniques means that thesauri are presently an important component of any practical multilingual text retrieval system, regardless of whether an exact match or a ranked retrieval model is adopted. Furthermore, integration of thesauri with techniques based on corpus statistics is an area of active research in computational linguistics, and there is some indication that the best features from each can be captured when the two techniques are combined [57]. Because the most sophisticated multilingual text retrieval thesauri in existence are in controlled vocabulary systems, ongoing research efforts would likely benefit from leveraging what has been learned in this work.

The differing domains of available parallel corpora and scored corpora (corpora for which relevance judgements are available) remains the largest single obstacle to evaluation of corpus-based techniques. We are not aware of a single instance of a large parallel corpus with an associated set of queries for which relevance judgements are available. Without such a corpus, the best possible experiment design is to train on a parallel corpus from a domain similar to that of the evaluation corpus. We are not aware of any existing techniques for estimating the the degree of a mismatch between the training and the evaluation domain or the effects of that mismatch. Without either scored parallel (or comparable) corpora, or at least some way of estimating the effect of a domain difference it will be difficult to draw conclusive conclusions from large-scale studies such as those conducted by Davis and Dunning [29]. We have developed a technique which provides some insight into this "domain shift" problem, and we describe

it in detail in Chapter 5.

The performance of monolingual techniques under identical experimental conditions appears to be a good benchmark for an upper bound on retrieval effectiveness. There is presently no evidence that multilingual techniques can reliably exceed the performance of monolingual techniques. Fluhr and Radwan have demonstrated that it is reasonable to lower bound the effectiveness of a multilingual text retrieval system with the effectiveness of a modular approach in which fully automatic machine translation is used to preprocess the query, and our analysis in section 4.2 supports this assertion. Agreement on these two common points of reference would facilitate comparison of multilingual text retrieval approaches across different experiments. The resources required to realize the potential of modern fully automatic machine translation systems may limit the utility of this approach in smaller studies, however.

One important difference between monolingual and multilingual retrieval is that polysemy appears to be a key limiting factor. In particular, polysemy seems to become a problem more rapidly in multilingual retrieval than in monolingual retrieval as the size of the domain increases. Three research groups (Radwan and Fluhr, Hull and Grefenstette, and van der Eijk), operating with very different experiment designs, have confirmed that polysemy can be reduced by indexing phrases rather than individual words. Since phrase indexing results in at best limited performance improvements in a single language [85], this leads us to conclude that the adverse effects of polysemy are considerably more severe in a cross-language retrieval and that techniques which reduce the number of possible translations serve to mitigate this effect somewhat. This suggests that word sense disambiguation, which has yet to reliably demonstrate even limited utility

in monolingual text retrieval [130], might be a productive avenue for further investigation.

The key issue in application of any natural language processing technique to multilingual text retrieval is to improve precision without a significant adverse effect on recall. This argues for investigating relatively shallow techniques that can be designed to degrade gracefully as the domain drifts. One of the pitfalls of translating queries is that short queries may increase the adverse effect of polysemy by limiting contextual clues about word sense. In order to deal with this effect, Hull and Grefenstette have proposed using structural information from the document space to enhance domain-specific interpretation of the query [69] and Radwan and Fluhr have implemented a simple version of this approach. In contrast, the vector translation technique that we develop in detail in Chapter 5 exploits the structure of user interest evidence gained over time. The two approaches seem complementary, with the decision between them depending on the relative rate at which the document space and the users' information needs are changing.

4.4 Summary

We have described a taxonomy of multilingual text retrieval approaches that is based on a fundamental division into knowledge-based and corpus-based approaches. Controlled vocabulary and concept retrieval are the two dominant knowledge-based approaches, although text translation offers a relatively straight forward modular solution. Deeper semantic processing has been applied in a few cases, most notably in the EMIR project. Automatic thesaurus construction

bridges the gap between the knowledge-based approaches which depend on thesauri designed for human use and the other corpus-based approaches which are designed only for automatic operation. The linear and nonlinear approaches to vector translation complete the taxonomy.

Of the knowledge-based approaches, only text translation and its better integrated counterpart, concept retrieval, offer the broad domain coverage required for many text filtering applications. We have selected text translation for our experiments in the next chapter because it is relatively easily implemented.

Latent Semantic Indexing offers an attractive corpus-based approach to adaptive multilingual text filtering because the LSI-mean adaptive text filtering technique is designed to exploit LSI feature vectors. We are particularly interested in comparing the performance of knowledge-based and corpus-based approaches to adaptive multilingual text retrieval, so we have chosen to evaluate a Latent Semantic Indexing approach as well.

We have chosen to also include a vector translation technique in our multilingual text filtering experiments because the vector translation technique we have developed is easily modified to take advantage of an existing bilingual term list. This provides a combination of the corpus-based and knowledge-based approaches that is similar in spirit to the approach used in Kikui's TITAN system. We describe our vector translation technique in detail in the next chapter.

The cross-language relevance feedback technique proposed by Fluhr and the automatic thesaurus construction technique implemented by Sheridan and Ballerini also offer other practical alternatives for adaptive multilingual text retrieval. Since both are similar in some ways to the Latent Semantic Indexing technique which we have chosen to evaluate, we felt that the three techniques

we have chosen provide sufficient breadth for the first experiments in adaptive multilingual text filtering.

Chapter 5

Adaptive Multilingual Text Filtering

In this chapter we build on the insights gained from our reviews of text filtering and multilingual text retrieval to develop and evaluate techniques designed to support the multilingual text filtering process. We assume the existence of a stream of texts in various languages, and desire to construct a profile which can be used for texts in any language that the system is designed to support. In our experiments we have investigated only the two language case, so we shall introduce our approaches in that context. All of the approaches can be generalized to cover multiple languages, and we have made some comments on the issues introduced by such a change in the next chapter.

The most straightforward solution to the problem is to create two separate profiles, one for each language. The multilingual filtering problem is then reduced to its well studied monolingual counterpart. This approach is quite reasonable for filtering systems like the Fast Data Finder described in Chapter 4 in which the profiles are constructed manually and automatic profile translation facilities are provided, particularly if expert assistance is available from trained intermediaries who possess expertise in the required languages and knowledge domains.

An uncoupled collection of monolingual filtering systems is not as well suited to adaptive multilingual filtering, however. Adaptive systems, systems capable of refining profiles based on experience, will be most useful when they can use information learned in one language to select documents written in another language. Returning to one of the examples in Chapter 1, a commodities trader might be following information about grain prices in Texas, principally reading documents in English. Should information on the same subject be published in Spanish, it would be useful if the system could use the profile generated from the trader's judgements about the English documents to select the documents in Spanish for display. It is this cross-language learning feature which distinguishes adaptive multilingual text filtering from its monolingual counterpart. We have thus chosen to adopt a single adaptive text filtering technique (the LSI-mean user model) and concentrate our experimental work exclusively on the cross-language aspect of adaptive multilingual text retrieval.

5.1 Techniques

We are not aware of any prior work on adaptive multilingual text filtering, but in Chapter 4 we identified several multilingual text retrieval techniques that have a clear potential for adaptation to support filtering applications. Both knowledge-based and corpus-based techniques can be used, and the choice between them might be expected to be based as much on the available sources of cross-language knowledge (e.g., thesauri or corpora) as on the relative performance of the techniques.

The simplest knowledge-based technique is to develop a thesaurus appro-

priate to the domain. But, because thesauri depend on restricted domains to minimize the adverse effects of polysemy (word sense ambiguity), a thesaurus-based approach is not well suited to broad-domain filtering. A more suitable alternative when performance in diverse domains is desired is to apply a broad-domain machine translation system.

The profiles used in adaptive text filtering systems are typically not natural language statements, so a straightforward implementation of the standard approach in multilingual text retrieval (translating the query) will often not be possible. The obvious alternative, is to translate every document into a single language. We call the adaptive text filtering technique we have developed using this approach "Text Translation," the first of the three techniques that we will evaluate in this section[1]

Although translating documents rather than queries is clearly less efficient, the relative efficiency will depend on details of the application. For example, applications in which the majority of the documents are already in a single language would not incur too large a penalty from translating only the documents that were not in that language. Furthermore, as with the text translation approach to multilingual text retrieval, some savings can also be obtained by eliminating unnecessary language generation components such as word order choice and unique translation choices if a vector representation will eventually be used.

The most serious impediment to the success of knowledge-based approaches such as Text Translation is that application of the technique in a new domain may require a considerable knowledge engineering effort. This is a particular

---

[1]We capitalize the name of the techniques in this chapter to distinguish them from the generic use of the same terms in Chapter 4.

problem in applications with frequent or severe domain shifts. The alternative to knowledge engineering is knowledge acquisition, and corpus-based techniques provide a way of automatically acquiring domain-specific knowledge of cross-language mappings from parallel text corpora.

Text Translation could also be used to evaluate a corpus-based approach, to adaptive multilingual text filtering by using a corpus-based (rather than knowledge-based) machine translation system. Statistical machine translation, in which translation probabilities conditioned on multi-term sequences that are developed using a parallel multilingual text collection and then used to predict the correct translation of each term, is one way in which this could be done [16]. Example-based machine translation, in which the translation of sequences of terms is guided by examples drawn from a parallel test collection, offers an alternative corpus-based machine translation approach [55].

Instead of evaluating two variations on Text Translation, we have chosen to evaluate a corpus-based cross-language mapping technique that can be more closely integrated with vector space text filtering approach. The cross-language LSI technique is clearly well matched with the LSI-mean user model that is described in Chapter 3, so we have chosen cross-language LSI as our corpus-based technique. Since the LSI-mean user model can be used with any cross-language mapping technique that produces a vector representation of text, the LSI-mean user model is compatible with Text Translation as well. The experiment design we present below takes advantage of that fact to produce results which compare the effectiveness of the two techniques under similar experimental conditions. Because we use the LSI-mean user model with all three of the cross-language mapping techniques, we refer to the cross-language application of LSI as "Latent

Semantic Coindexing" in order to minimize the potential for confusion.

Latent Semantic Coindexing is by no means the only practical cross-language mapping technique that can be closely integrated with vector space text filtering. The relevance feedback multilingual text retrieval technique described in Chapter 4 integrates a technique similar to example-based machine translation closely with the vector representation, relying on explicit examples to find a translation. Sheridan and Ballerini's corpus-based query expansion technique uses a more sophisticated representation, but both techniques essentially rely on the same idea—finding a prior example which is sufficiently close to the terms being indexed. Any one of the three techniques could be used for multilingual text filtering filtering when parallel bilingual training corpora are available. The experimental results for Latent Semantic Coindexing that we report below are simply the first step in evaluating this range of possibilities.

The availability of parallel bilingual corpora is an important prerequisite to the use of any corpus-based technique, and one which can not be passed off lightly. While the advantage of corpus-based techniques is that they can potentially acquire knowledge of how the languages in question are used in a particular domain, this domain-specificity can be a liability if the domain of the parallel training corpus differs significantly from the application domain. Thus, corpus-based techniques will be most useful when high-quality translations are produced as a natural byproduct of ongoing activity that is closely related to the multilingual text filtering application. An example of such a situation would be international negotiations, in which there is a continuing need for generating high quality translations, and a related interest in developing and maintaining an awareness of news articles related to the subject of the negotiations, regardless

of the language in which those articles appear.

One way of expanding the applicability of a corpus based technique is to use domain knowledge to constrain the allowable representations. This idea, a combination of the knowledge-based and corpus-based approaches, offers the potential for performance superior to that which could be achieved by either approach in isolation. A comparison of Text Translation and Latent Semantic Coindexing can be expected to yield some insight into the relative performance of the knowledge-based and corpus-based approaches, but it will be silent on how the two approaches might be combined. We have designed our third technique, Vector Translation, to exploit both statistical information and manually encoded knowledge in order to begin to explore this issue. Our Vector Translation technique was inspired by earlier work in statistical machine translation, but the simplicity of the distributions which are learned from the training corpus makes the approach amenable to knowledge engineering both before and after the statistics are collected.

The next three sections describe Text Translation, Latent Semantic Coindexing and Vector Translation in detail. In the remainder of this chapter we describe the design of our experiments and the test collections that we use, present results comparing the effectiveness of the three techniques, and draw some conclusions about their relative merits.

5.1.1 Text Translation

The approach we call Text Translation (TT) involves using knowledge-based machine translation to bring each document that will be used to construct the profile into a single language and then building the profile using the LSI-mean

technique. Subsequently arriving documents are then translated (when necessary) into that same language and the the LSI-mean technique is used to rank those documents. In our experiments we use a broad-domain English-to-Spanish machine translation system provided by the Logos Corporation, so we have chosen English and Spanish as our experimental language pair and we use Spanish as the single language into which documents in English must be translated.

Figure 5.1 shows a typical TT design for this language pair. The architecture is quite general, since any machine translation system can be inserted and any vector-based adaptive text filtering technique (e.g., relevance feedback, LSI-mean, the Gaussian User Model, a neural network or a probabilistic technique) can be used in the profile learning module. The initial profile can be generated in a number of ways. For example, it could be provided directly by the user, it could be automatically constructed from a "typical" set of desirable documents provided by the user, or it could be selected automatically from a predefined set of stereotypical profiles based on responses to a screening questionnaire. Regardless of how the initial profile is acquired, it is simply passed through the profile learning module to the selection module. A separate profile is maintained for each distinct information need.

English documents are submitted to the machine translation system upon arrival, and the Spanish translations are used to construct vectors based on the frequencies with which terms appear in those documents. Spanish documents bypass the machine translation step and are used directly to produce vectors. The indexed phrases may be based on phrases, word stems, the morphological root forms used internally in the machine translation system, or character sequences of arbitrary length (e.g., bigrams or trigrams). Term weighting

167
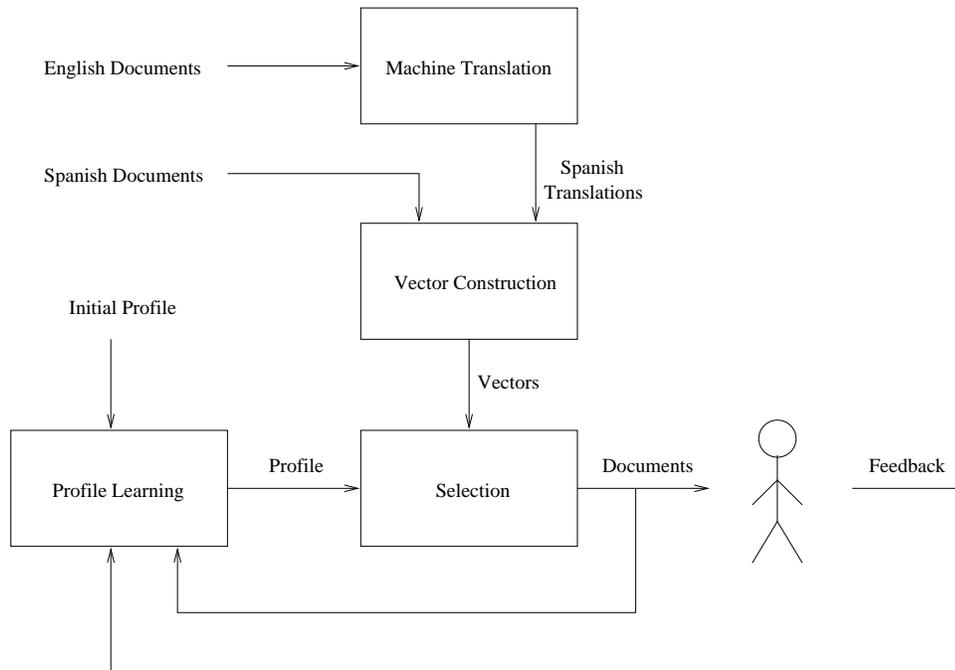
Figure 5.1: Text filtering using Text Translation.

functions which require collection-wide statistics such as "inverse document frequency" must estimate those statistics because the collection itself is dynamic. In general, periodic updates to collection statistics can be based on a sample of the most recent documents. The parameters of this "sliding window" approach (e.g., sampling technique, window size, and update periodicity) can be adjusted to balance time and space efficiency with the rate of change in term usage in the specific application.

The selection module uses the profile and the document vectors to select and arrange documents for display to the user. This may involve set selection, ranked output, or some more advanced visualization strategy. In other words, this "selection" module combines aspects of the "selection" and "display" components in Figure 2.1, a simplification we will retain for the remainder of this chapter. Although the selection module makes use of the vectors, it is the orig-

168

inal documents that the selection module provides to the user. The association of the vectors to the documents must be maintained by the machine translation and vector construction modules. This too is an implicit assumption we shall consistently make in the remainder of this chapter.

The user's responses to some subset of the documents are then observed by the system and provided as feedback to the profile learning module. This feedback may be based on explicit user reactions (e.g., "like it"/"hate it," or some ordinal preference scale) or it might be based on implicit feedback—observations of behavior for which some relation to user preferences is known (e.g., reading time or selection of documents from a menu listing their title and author). The profile learning module must also know with which document each reaction is associated, and we show that feedback loop explicitly. For vector-based user models such as the LSI-mean technique it is sufficient to pass the document vector, rather than the document itself, to the profile learning module.

The Text Translation architecture is thus quite straightforward, adding only a machine translation component to what would be found in a typical monolingual vector-based text filtering architecture. Although machine translation errors can be expected to reduce effectiveness somewhat when processing documents which must be translated, use of a high-quality translation system appropriate to the knowledge domain of the documents can minimize this source of error. The effectiveness of the machine translation system must, however, be balanced against it's efficiency. For applications in which a substantial number of documents require translation, the throughput of the machine translation system is likely to be a limiting factor. The Logos machine translation system we use in our experiments is able to translate approximately 100,000 words (about 500 pages)

per hour on a SPARC 20. While this is perfectly adequate for our experiments (which are conducted without user interaction) high volume applications would require real-time translation capabilities.

5.1.2   Latent Semantic Coindexing

Text Translation requires that every term frequency vector be constructed using documents that have been translated into a single language, but Latent Semantic Coindexing (LSC) produces a language-independent representation directly from the original documents. Figure 5.2 shows a typical LSC system architecture. English and Spanish documents are provided directly to the vector construction module, along with what we call a "translation matrix," the T matrix described in Chapter 3. This T matrix is produced by first performing a Singular Value Decomposition on a term-document matrix X constructed from a set of bilingual documents, each of which consists of English and Spanish versions of the same document. The first k columns of the T_0 matrix in the decomposition $X = T_0 S_0 D_0^{\mathrm{T}}$ are then used as the columns of T. As described in Chapter 3, values of k between 100 and 400 typically provide good performance, with the larger values being appropriate for larger collections in which a greater number of concepts need to be represented. We used $k = 200$ for the experiments reported in this chapter.

Recall that in LSI the first part of each row of the T_0 matrix is thought to encode information necessary to situate distinct concepts, while the last part is thought to encode the effect of term usage variations. In bilingual documents, the term usage variations which are suppressed include cross-language term usage variations as well. The vectors constructed by LSC thus serve as language-
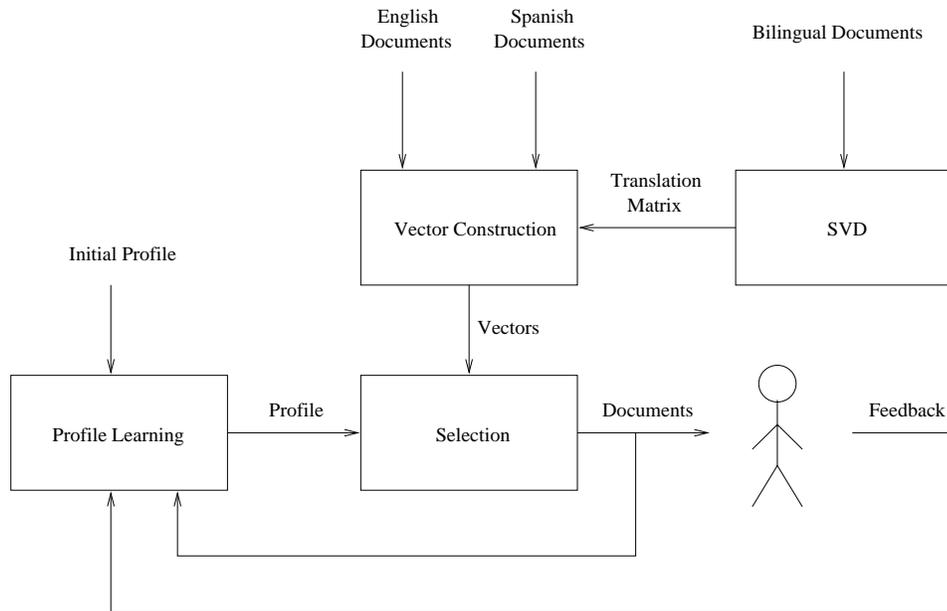
Figure 5.2: Text filtering using Latent Semantic Coindexing.

independent representations of the concepts contained in those documents [80].

LSC can in principal be used with a single static set of bilingual documents, but performance could suffer if the term usage pattern in the documents being filtered were to drift or shift away from the pattern in the bilingual document collection. Updated collections of bilingual documents may not be available in every application, but when such resources are available a sliding window approach similar to that used for computation of collection-wide term weighting factors can be employed.

In the vector construction module, vectors are formed in the usual way, including the computation of term weights using whatever weighting function is desired. It is important, however, that this be the same term weighting function that was applied to the term-document matrix before computing the SVD. The vectors are then augmented with a null vector for the terms in the second language and multiplied by the T matrix to produce a language-independent

concept vector for each document. Those concept vectors are then used by the selection and profile learning modules in the same way as described above for the TT technique

LSC is capable of processing documents considerably faster than the machine translation component of TT, but computing the SVD of a large term-document matrix can require enormous amounts of time and memory. The simplest way of limiting this computational complexity is to select a sample of representative documents from the bilingual document collection, and to arbitrarily limit the length of these documents. The length limitation has a beneficial side effect, because term usage patterns in shorter documents seem to provide a better basis for separating the effects of conceptual and term usage information using LSI [162]. In our experiments we compute the SVD on nearly 500,000 words from over 25,000 documents in about an hour on a SPARC 20, so periodic recomputation would certainly be practical on a collection of that size. Somewhat more efficient techniques for approximate incremental recomputation of the SVD can also be applied between complete recomputations to achieve additional savings [10].

5.1.3   Vector Translation

The knowledge acquisition step in LSC is guided by a single heuristic, that cross language term usage variation is encoded by the last columns of the T_0 matrix. This stands in stark contrast to the TT approach, in which all domain knowledge must be embedded in the text translation system, a task typically done manually. In our experiments, the middle ground—in which automatic knowledge acquisition is guided by metaknowledge encoded in the acquisition module—is filled by a technique we have developed that we call Vector Translation (VT).

Unlike TT and LSC, the VT technique is not inspired by an existing multilingual text retrieval technique. For that reason, we will describe its motivation and operation in some detail.

Like TT, in VT every document is used to produce a Spanish vector. But in VT it is the document vector, rather than the document, which is translated. VT is essentially term-by-term translation applied to the vectors which represent documents.[2] Since each element of a document vector is associated with a single term in a single language, term-by-term translation can be applied to vectors as easily as it can be applied to documents. Document vectors typically encode no word-order information (except, perhaps, when phrases are encoded as a single term), so deeper analysis is precluded by the representation. Term-by-term translation is quite fast, but (except in narrow domains where polysemy effects can be suppressed), the resulting translations are usually of extremely low quality. The reason for this poor performance is that without context it is impossible to determine what word sense was intended for polysemous words.

Fortunately, the vector representation has two features which mitigate the adverse effects of this problem. The first is that it is not necessary to select a single translation target for each term, since the vector representation is based on the frequency with which a term occurs. For example, if there are two possible Spanish translations for some English term, it would be possible (although perhaps not wise) to simply divide the weight associated with the English term equally to produce a weight for each of the Spanish terms. Figure 5.3 illustrates

---

[2]VT must be applied to the columns of the term-document matrix and not to the LSI feature vectors since the elements of the LSI feature vectors have no obvious relationship with individual terms.

this for two senses of the English word "bank," one of which (a financial institution) translates to "banco" and the other (a river bank) translates to "orilla."
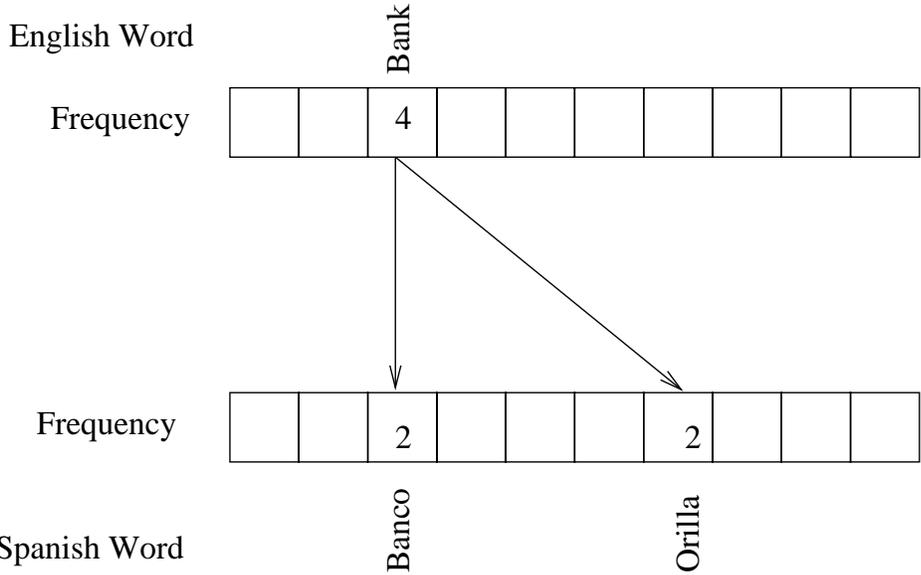
Figure 5.3: Division of an English term weights between two Spanish terms.

The second helpful feature of vector representations is that a kind of "reverse polysemy" effect reduces the adverse impact of associating some of the weight with the wrong term in the other language. Here it is helpful to consider the case in which two English terms both translate to the same Spanish term. The weights contributed by each English term can simply be added together to find the weight that should be assigned to the Spanish term. Word choice variation is a common stylistic device in many types of documents, typically introduced to avoid the monotony of repeated use of a single term. When different English terms are used for the same concept, it is likely that the intersection of their Spanish translations will be quite small, perhaps even a single word. Thus, term weight will tend to accumulate on the "consensus translation," and that

174

consensus is likely to be correct. Figure 5.4 illustrates this consensus translation
effect for the English terms "credit union" and "bank."
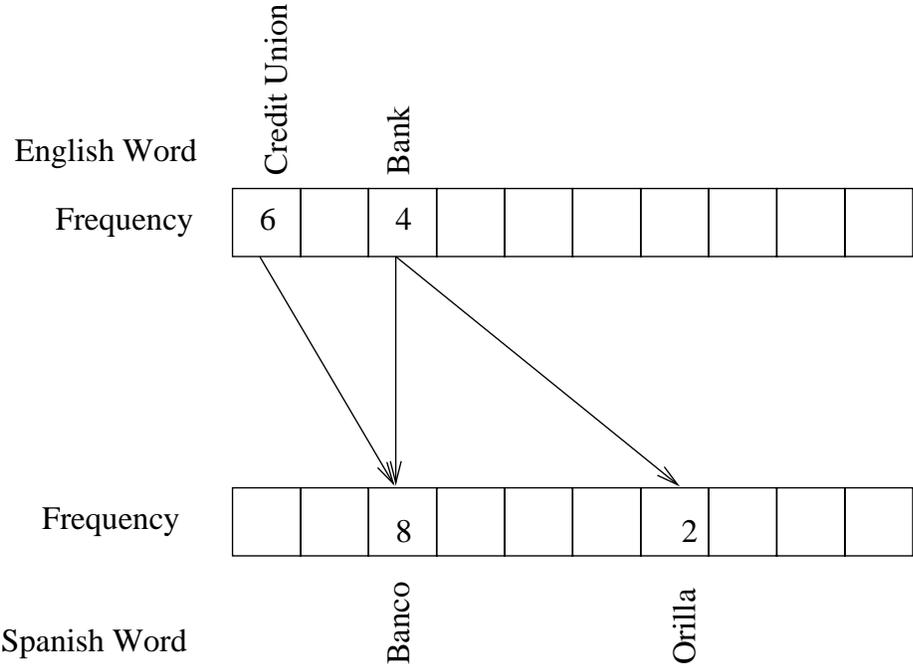
Figure 5.4: The consensus translation effect for Vector Translation.

A useful way to consider these effects is to view each document vector as a
discrete distribution. In this context, term weight functions estimate the proba-
bility mass which should be assigned to a term (although the normalization factor
differs from that which would be needed to achieve unit length). Each document
is thus represented by the discrete distribution on the set of all possible terms.
The term weight splitting shown in Figure 5.3 is achieved by using a second
type of discrete distribution, a distribution (which is correctly normalized) on
the possible translations of each term. The collection of such distributions is rep-
resented by the translation matrix, and the mapping from Spanish into English
is achieved by multiplying the translation matrix and the English term vector to
produce a Spanish term vector. Since the translation matrix represents a linear

operator, the additive behavior necessary for the consensus translation effect is a natural consequence of this approach. Figure 5.5 illustrates this computation.
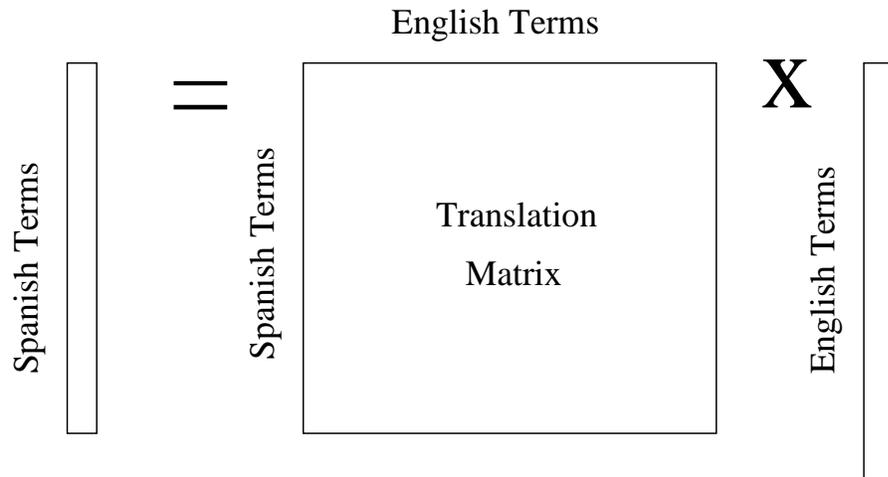
Figure 5.5: Application of the translation matrix in Vector Translation.

Figure 5.6 shows how this translation matrix is used for text filtering. Separate modules are used to construct vectors for English and Spanish documents because term lists and collection-wide statistics differ for the two languages. Vector translation is performed for vectors based originally on English documents so that all of the vectors passed to the translation module approximate those which would have been constructed if the original documents had been in Spanish. The remainder of the filtering process then proceeds as in the other two techniques.

Vector Translation would not be expected to work well on the short queries commonly found in many text retrieval applications because there would be little or no opportunity for the consensus translation effect to develop. But the technique seems well suited to multilingual text filtering since relatively long documents must be translated anyway. In the Text Translation technique, the requirement to translate long documents is a disadvantage because it is time con-
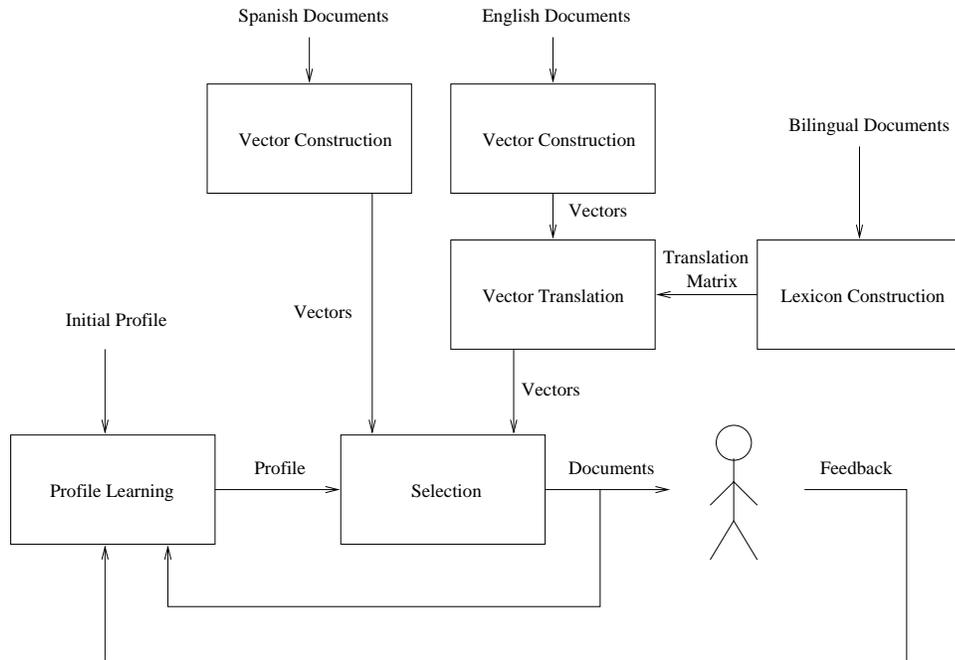
Figure 5.6: Text filtering using Vector Translation.

suming. Once the translation matrix has been constructed, Vector Translation can be applied as quickly as Latent Semantic Coindexing, and the length of the documents is an advantage because it provides greater scope for development of the consensus translation effect. The Vector Translation technique might also be useful in a multilingual text retrieval system when performing relevance feedback or when performing "query by example" in which a sample document is offered as the query.

Even under the best of circumstances, a translation matrix is bound to introduce some sources of error. Matrices represent linear functions, and a linear function can capture some aspects of the consensus translation effect. But intuition suggests that the optimal encoding of the consensus translation effect would probably require a nonlinear function. For example, in Figure 5.4 it would probably make more sense to place all of the term weight from "bank" on "banco"

since two other English terms translated to "banco" and no other terms in the document translated to "orilla." Using a linear approximation to the optimal vector translation function will result in a somewhat more diffused distribution for the translated vector, with some of the term weight distributed to useless (and probably even counterproductive) terms.

Of course, Text Translation and Latent Semantic Coindexing each introduce their own types of errors. In Text Translation the principal sources of error are failure to recognize a Spanish word and incorrect resolution of polysemy. In Latent Semantic Coindexing the principal source of error is the inability to separate polysemous uses of a term in the training collection, which makes it impossible to conflate terms that have different sets of polysemous senses across languages into a single concept representation. In Vector Translation the error introduced by the restriction to a linear mapping is likely to be dominated by errors in the construction of the translation matrix itself, a process which we describe in detail below. The principal goal of our experiments has been to compare these three techniques in order to determine which sources of error have the smallest adverse impact on the performance of an adaptive multilingual text filtering technique.

Actually, both LSC and VT use a translation matrix, so both are subject to the limitations imposed by linearity. Figure 5.7 illustrates the corresponding computation for LSC. Mathematically the two operations are nearly identical, differing only in the number of elements in each vector and matrix. But in practice, the LSC translation matrix maps from sparse English or Spanish document vectors to a dense concept vector, while the translation matrix in VT always maps from sparse English document vectors to sparse Spanish document

178

vectors. Vector translation therefore is more similar in practice to Text Translation, since a preferred language in which every document will be represented must be selected. For our experiments, this preferred language is Spanish. Of the three techniques we have evaluated, LSC alone develops a language-independent representation for each document.
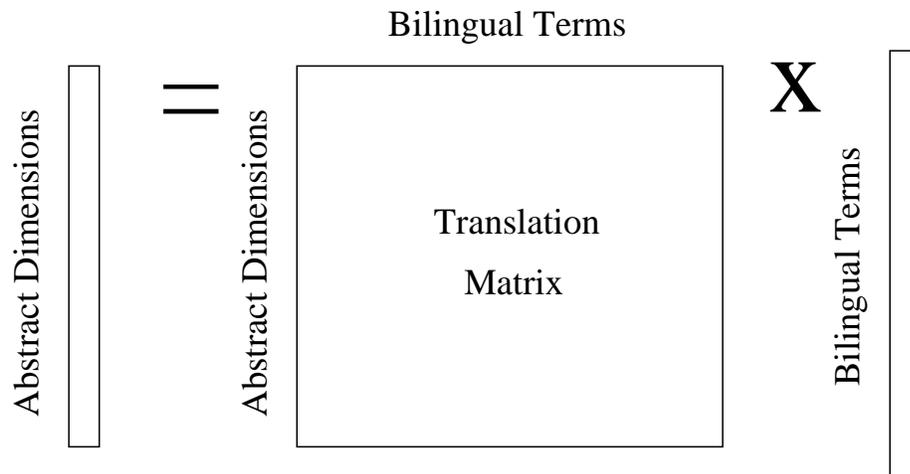
Figure 5.7: Application of the translation matrix in Latent Semantic Coindexing.

The other key difference between LSC and VT is the source of the information on which the translation matrix is based. In LSC this information is collected automatically from documents which are aligned only at the document (or passage) level. Because the individual components of the T_0 matrix lack any understandable individual interpretation, human assistance with the construction of the T_0 matrix is precluded. The VT translation matrix, on the other hand, is constructed from bilingual document collections in which individual terms have been aligned. Several techniques have been proposed for performing this alignment (c.f., [17, 155]), and we describe the one we have chosen below. An empirical distribution is then constructed to find the probability that an English word will map to each possible Spanish word. In addition to exploiting

a somewhat different source of information, this approach gives the elements of the VT translation matrix a natural interpretation. Each is the probability that a specific English word will be translated to a specific Spanish word for documents in the domain of interest. Such information can be collected automatically from bilingual document collections, but it can also be corrected and augmented using lexical information produced by humans. Since corpus-based and knowledge-based systems might be expected to make different types of errors, this joint construction approach could result in improved performance.

Term alignment in bilingual document collections is a challenging problem which is studied in the field of corpus linguistics, a branch of natural language processing. Term alignment typically proceeds in three stages:

1. Document alignment. Corresponding documents in each language are identified.

2. Sentence alignment. Sentences and other similar units in each text are identified and corresponding pairs of sentences are associated.

3. Term alignment. Corresponding terms (which may be words, word stems, morphological roots appearing in a dictionary, and/or multi-word phrases) in aligned sentence pairs are identified.

In our experiments we use a parallel bilingual collection of United Nations documents that has been aligned at the document level by the Linguistic Data Consortium at the University of Pennsylvania using document numbers assigned when each document was originally prepared. David Hull of the Rank Xerox Research Corporation has used proprietary software to preprocess each document used in this experiment, convert each word to it's morphological root and sub-

stituting a unique token for commonly appearing phrases.[3] Shen and Garman have developed a statistical technique which uses dynamic programming to optimize the alignment of sentences based on their length [135]. Wade Shen at the University of Maryland has applied that technique to align the preprocessed documents in the UN collection, and we have used the results of that alignment in our experiments.

Shen is presently extending this software to perform term alignment as well [135]. The technique is based on the cooccurrence frequency of terms in aligned sentence pairs, with greater weight placed on cooccurrences that appear at similar locations within each sentence. For example, two words will be assigned a greater cooccurrence value if they both occur as the first word in a pair of aligned sentences than if one appears first and the other appears last. Every term pair with a cumulative cooccurrence value that exceed a specified threshold is considered to be aligned. The number of such cooccurrences is then used to compute an empirical distribution on the Spanish translation of every term appearing in the English versions of the United Nation documents, and those distributions are stored as a translation matrix.[4]

It is the use of a threshold on the correlation values which induces some measure of term alignment, and that is what distinguishes this approach from the vector translation technique of Davis and Dunning described in Chapter 4 which was based solely on sentence alignment. A high threshold results in a sparse translation matrix and highly focused vector translations, a lower thresh-

---

[3]When phrases are recognized the words which comprise the phrase are removed from the text and only the unique token is provided.

[4]It is important to note that the translation matrix is based on the number of alignments for each term, and not on the correlation values.

old produces more translation targets for each term and hence a somewhat more diffused translation in which the same amount of term weight is spread across a larger number of terms. One of the goals of our experiments will be to determine the threshold value which produces enough diffusion to exploit the consensus translation effect, but that remains focused enough to avoid introducing an overwhelming number of spurious translations.

Other techniques for term alignment have been proposed as well. Brown, *et al.* at IBM have collected term translation statistics using vastly more sophisticated techniques which directly handle word to phrase translation and take advantage of information encoded in word order [17]. The result is a translation matrix which is conditioned on sequences of English words rather than on a single word. Such a distribution is easily converted to one conditioned on the final term in the sequence by summing across the possible prefixes of that term, although it is not clear whether the result would be any more accurate than Shen's simpler technique.

Shen's technique is similar to the automatic thesaurus construction technique used by van der Eijk that we described in Chapter 4 [155]. The principal difference between the two is that Shen's technique incorporates provisions to generate the empirical distribution.[5] In fact, it is useful to think of the translation matrix in Vector Translation as a kind of "stochastic thesaurus" in which each possible translation is labeled with the probability that that translation will occur in the training collection of bilingual documents.

This observation suggests two ways in which human knowledge can be used

---

[5]The other difference is that Shen's technique presently does not rely on part of speech information.

to guide the development of the stochastic thesaurus. Many of the candidate alignments produced by statistical techniques such as Shen's make no semantic sense. But bilingual dictionaries typically seek to list every possible translation, and often those translations are listed in some sort of preference order. It should be possible to use the set of known translations in an existing bilingual dictionary as a stronger constraint on the alignment process than the present threshold on the cumulative cooccurrence value. We call this technique "seeding" the distribution with the dictionary since the probability mass is constrained to accumulate only on the seeds that we have provided. It may also be possible to exploit the order of appearance in the dictionary entry as an additional constraint on the relative size of the probability masses that are assigned to the translations, although we have not yet designed an approach for doing so. While seeding the distribution tends to drive the translation matrix from one tailored to a domain towards one suitable for more general application, the improvement in alignment accuracy (and hence in the effectiveness of the VT technique) could be significant. We have not yet implemented seeded vector translation, but it is the potential to integrate both knowledge-based and corpus-based techniques in this way that motivated the choice of Vector Translation as our third technique.

The other way of adding human knowledge to the translation matrix is by hand-tuning the matrix after it has been constructed. If analysis reveals unusually poor performance for the cross-language component of a Vector Translation multilingual text filtering system on a particular set of topics, the translation probabilities for keywords associated with those topics could be examined by a domain expert who is fluent in both languages. If the values in the matrix appear to be counterintuitive, it would be possible to adjust them manually. Such a

process is not likely not prove economically feasible for many applications, however, unless automated tools can be developed which can identify potentially poor translation probabilities and either suggest improvements or apply those improvements without human intervention.

5.2   Experiment Design

The principal objective of our experiments is to compare the performance of three techniques, TT, LSC and VT. Comparison of knowledge-based and corpus-based techniques is inherently difficult, because each is designed for a different application environment. Furthermore, because each technique requires unique design decisions, it can be difficult to generalize from the results of experiments run with specific parameter choices. Finally, we are not aware of any test collection that combines all of the features that are useful for evaluation of multilingual corpora. In this section we describe a set of experiments which are designed to produce useful comparisons in the face of these challenges.

In order to produce comparable results, our basic strategy is to establish experimental conditions that are as similar as possible across the three techniques. We use the LSI-mean filtering technique in every case, because LSI is already an integral part of the LSC technique [41]. Because we envision applications which must be able to filter bilingual collections of monolingual documents, we must introduce a separate collection of bilingual documents on which we can perform the bilingual SVD for the LSC technique. For consistency, we use the same collection of United Nations documents for this purpose that we have used to construct the translation matrix for the VT technique. To further assure con-

sistency, we perform the monolingual SVD required for the LSI-mean technique using the Spanish version of the documents in that collection.

Since we are presently interested only in characterizing cross-language filtering performance, we have chosen to train the LSI-mean profile using documents in one language and then evaluate filtering effectiveness using only documents in the other language. The LSI-mean filtering technique exploits only positive training examples, so we have chosen to develop the profile using documents in English and then evaluate it's performance using documents in Spanish. This choice minimizes the number of documents which must be translated from English to Spanish for the TT evaluation. It should be noted, however, that this choice minimizes only the difficulty of running our experiment. In practical applications, every arriving document would need to be brought into Spanish in order to apply the TT technique. We achieve this simplification for the purposes of our experiment by measuring the performance of the system at a snapshot in time—after training on one set of documents, we conduct an evaluation on a second set. Because we know *a priori* which documents in the training set are relevant to each topic, we can avoid unnecessary translation of all the rest of the training documents. And because the entire evaluation set is already in Spanish, no additional translation resources are required for evaluation.

Substitution of word stems for surface terms is often used to improve the effectiveness of a text filtering system. Instead we have chosen to consistently substitute the morphological roots and phrase tokens provided by David Hull at the Rank Xerox Research Centre for the surface terms because those root forms and phrase tokens are used to produce the VT translation matrix. We remove common words using a bilingual stopword list formed by conjoining the English

and Spanish stopword lists provided with SMART version 11.0. This results in occasional removal of terms in one language by the stopword list for the other language, but this effect is not significantly more severe than the occasional removal of a content-bearing word in a monolingual context.[6]

Figure 5.8 shows the three step top-level design that we have used for all of our experiments. For LSC the "LSI training collection" consists of the bilingual UN documents, while for TT and VT it consists of the Spanish versions of those same documents. The output of the SVD step is the T matrix and a term list which relates each row in the T matrix to a single morphological root. Table 5.1 shows the parameters used in the SVD step.

Parameter	Value
Dimensions Retained (k)	200
Stopword List	Bilingual
Term Weight Function	SMART ltc weights

Table 5.1: Experimental parameters for the SVD step.

The "profile training collection" must contain both English documents and binary (relevant/not relevant) relevance judgements for a number of topics. These documents and relevance judgements are used by the LSI-mean technique (in conjunction with the T matrix and the term list) to produce a concept vector which represents the profile. The term list is fixed by the SVD step, so terms which appear in relevant documents of the profile training collection but do not appear in the term list must be ignored because there will be no corresponding row in T. Documents for which no relevance judgements are available are also

---

[6]For example, "c" is included in SMART's English stopword list, an unfortunate choice when seeking to identify documents about computer programming languages."
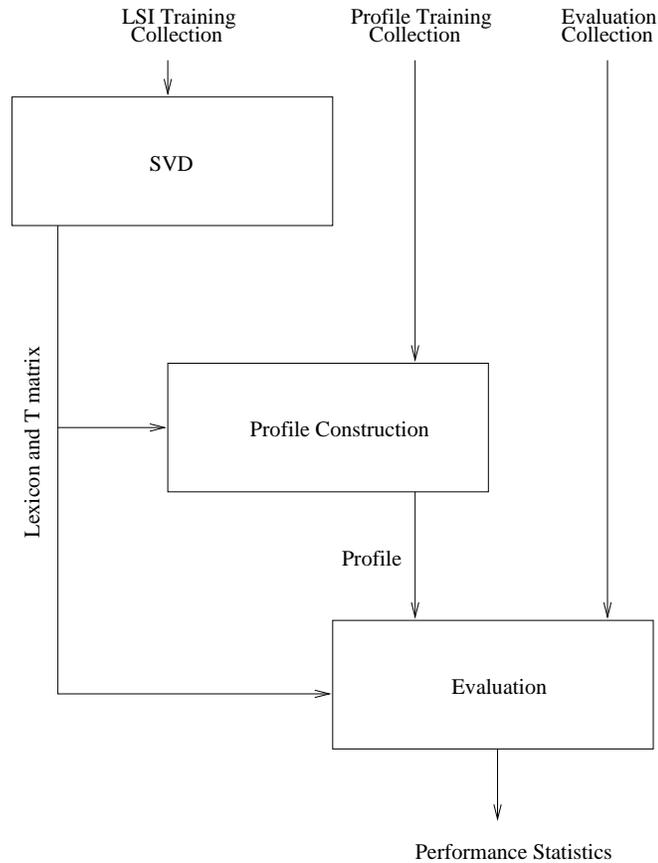
Figure 5.8: Top-level experiment design.

ignored (i.e., they are treated as if they are not relevant). The SVD step need only be performed once for each type of experiment (LSC or TT/VT), but the profile construction step must be repeated for each topic.

The evaluation step must also be repeated for each topic and, of course, for each of the three techniques. The "evaluation collection" must consist of Spanish documents for which relevance judgements are available, and those relevance judgements must be made with respect to the same collection of topics that were used to make the relevance judgements for the training collection. Concept vectors for each Spanish document are first computed using the T matrix and the term list provided by the SVD step, again ignoring terms which do not appear

in the term list. The Spanish documents are then ranked in order of decreasing similarity with the profile concept vector using the cosine measure. Finally, the precision at a recall level of 0.1 is computed to produce a topic-specific figure of merit that is appropriate for interactive applications in which high precision is preferred over high recall. If desired, these figures can be averaged over several topics to produce a single figure of merit which describes the performance of each technique on this set of test collections.

Each step is implemented using the same modified version of the SMART experimental text retrieval system that was used for the experiments described in Chapter 3. The changes we have made to SMART, including the additional modifications to implement the Vector Translation technique, are described in Appendix A. A value of $k = 200$ was consistently used for the experiments reported in this chapter because that value was found to result in good performance with monolingual evaluations on the test collections we have chosen.

5.3   Test Collections

Corpus based techniques such as LSC require bilingual document collections with the same structure of language use as is found in the profile training and evaluation collections. We thus would be able to gain the greatest insight into the relative effectiveness of our three techniques if some of the documents in the United Nations (UN) collection could be used for LSI training, others for profile training, and the remainder for evaluation, with the documents randomly assigned to one of three groups in order to minimize the sources of experimental error. Because significant differences in filtering effectiveness occur across topics,

the results will be most representative of "typical" performance when averaged over a fairly large number of topics, so the ideal test collection would also have a large number of topics against which every document (or at least every document that is not assigned to the LSI training collection) has been judged.

The UN collection certainly fails to meet those requirements, because there is no standard set of topics associated with it and hence there are no relevance judgements at all. But since we are not aware of any large parallel bilingual document collection which satisfies these criteria, the UN collection provides a reasonable starting point. A set of topics appropriate to the issues addressed in that collection could be constructed manually, and relevance judgements assigned to documents with respect to each topic. Relevance judgements are expensive to collect, however, so sampling techniques such as the pooled assessment methodology used in TREC would be needed. For experiments conducted when the pool is established, the pooled relevance assessment methodology produces exact values for precision but can only compute an upper bound on recall. In order to ensure repeatable results on a standard set of relevance judgements, subsequent experimenters will often treat unassessed documents as if they were known to be irrelevant. This will have almost no effect on the actual effectiveness of filtering technique such as LSI-mean which are based only on relevant documents, because only a small number of potentially useful training documents are likely to be missed. Both recall and precision will only be approximated when this technique is used for evaluation, but the results can be compared to other experiments which have used TREC relevance judgements after the pool was created.

Clearly such an extensive effort cannot be justified for a single set of exper-

iments. The obvious alternative, translation of an existing collection for which relevance judgements are available, is unfortunately even less practical. If the results of the evaluation are to be representative of documents which occur naturally in different languages, the required translations would be of a higher quality than can be achieved with the present technology for fully automatic machine translation. Although some efficiencies could be achieved by limiting translation requirements for the profile training collection, there would be no alternative to translating all of the documents used in the LSI training step.

This situation poses somewhat of a quandary. Both approaches for constructing a test collection would require a substantial commitment of resources, but such resources are unlikely to be made available until there is some evidence that practical multilingual filtering techniques can be constructed. The approach that we have taken to resolve this dilemma it is to relax some of the requirements we have stated for an "ideal" test collection and then seek to characterize what has been lost in the process. Although the actual test collections we have used in our experiments lack many of the desirable characteristics we have identified, they do provide sufficient insight into the performance of the TT, LSC and VT techniques to determine whether committing the required resources to develop test collections better suited to multilingual text filtering evaluation can be justified.

We have chosen to use three existing collections for our evaluation. As we described above, the Linguistic Data Consortium has collected a large parallel corpus of United Nations documents which contains over 1 GB of documents in three languages: English, Spanish and French. We use the English and Spanish documents that were generated during 1992 as the LSI training collection because memory requirements of the SVD for resulting term-document matrix

were compatible with the capabilities of our computing resources. Some of the UN documents are quite long, and the experience of others that we described in Chapters 3 and 4 has led us to conclude that the performance of LSI can be significantly improved by constraining the length of the documents [132, 162]. The characteristics of the UN documents makes alignment at the end of the document simpler than alignment at the beginning, so for each English document we extract the last 400 tokens produced by the morphological analyzer (i.e., words, phrases, punctuation, and SGML tags) and a corresponding number of tokens from the associated Spanish document. Spanish documents from this source typically contain more words than their English translation, so we compute the expansion ratio for each document and expand the window at the end of the Spanish document by that factor. Only words and phrases are used to construct the document vectors because punctuation, SGML tags and stopwords are automatically removed in subsequent processing. A typical bilingual document vector constructed in this way contains nonzero term weights for about 200 unique terms, slightly more than half of which are Spanish words or phrases. For consistency, the expanded version of each Spanish UN document is also used to compute the SVD for the TT and VT techniques.

Although shorter documents seem to improve the performance of LSI, the term alignment software actually benefits from longer documents. Sentence pairs which cannot be aligned are simply ignored by the sentence alignment software, and terms which produce no above-threshold alignments will produce no translations. Hence, we chose to use the entire UN documents rather than the last 400 tokens when computing the translation matrix for the VT techniques. Furthermore, we used all of the UN documents written between 1990 and 1992 in

order to produce the best possible translation matrices.

For the profile training collection we use English newspaper articles from 1990-1992 that appeared in the U.S. newspaper the Wall Street Journal (WSJ). We selected this collection because it is from roughly the same period as the UN collection and because TREC relevance judgements (using a pooled relevance assessment methodology) are available for 300 topics for that collection. Manual inspection of these 300 topics reveals that four of them appear to be similar to (but not identical to) topics for which relevance judgements are available on the evaluation collection we have chosen.

For the evaluation collection we use Spanish newspaper articles from 1992 that appeared in the Mexican newspaper El Norte. TREC Relevance judgements are available for 50 topics on that collection, and four of those 50 are the topics we found similar to the topics against which the WSJ articles were judged. Table 5.2 identifies the four topic pairs, and Figure 5.9 shows our impression of the degree and type of overlap of the topic descriptions in each topic pair using Venn diagrams. Appendix B.2 provides additional details about the TREC collections we have used and the complete text of the four topic descriptions.

Spanish Language Topic		English Language Topic	
SP10	Mexican Narcotic Traffic	022	Counternarcotics
SP22	Mexican Inflation	008	Economic Projections
SP25	Mexican Privatization	128	Privatization of State Assets
SP47	Mexican Cancer Research	123	Carcinogen Research & Control

Table 5.2: Closely related English and Spanish TREC topics.

By using training and evaluation collections from the same periods, we have attempted to minimize the effect of temporal shifts in the content of the three
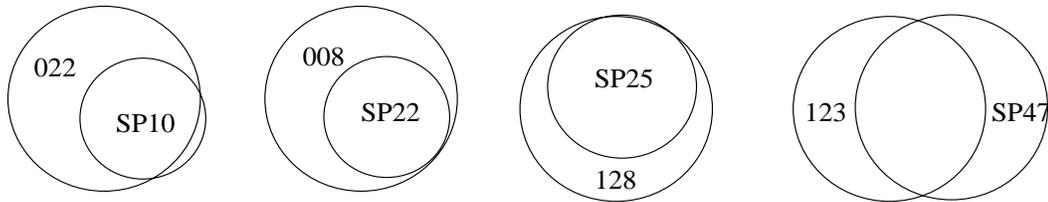
Figure 5.9: Topic overlap.

collections. For example, documents about UN sanctions on Iraq appear in all three collections because those sanctions were imposed in that time frame. A closer inspection of the three collections does, however, reveal some significant differences in the predominance of the topics in each collection. Many of the United Nations documents, for example are resolutions regarding sanctions on Iraq. The dominant theme of the Wall Street Journal articles is economic activity, and documents which discuss other issues often describe the economic consequences of those issues as well. As would be expected, a large number of the documents in the El Norte collection describe events in Mexico. In particular, sports such as soccer receive a good deal more coverage in El Norte than in either of the other two collections. We refer to the effect of these differences on our measurements of filtering effectiveness the "domain shift effect."

All three collections include at least some coverage of the four topic pairs in Table 5.2. So our expectation before conducting the experiments was that comparisons of effectiveness measures such as the recall at a precision of 0.1 would yield some insight into the relative effectiveness of TT, LSC and VT despite the differences in coverage. Davis and Dunning used two of these three collections for multilingual text retrieval experiments, however, achieving values of average precision close to that which would have resulted from random selection of documents [29]. With this in mind we have designed and conducted additional

193

experiments to characterize the magnitude of the domain shift effect in order to help interpret our results.

In addition to the domain shift effect, our experiment design also results in what we call a "topic shift effect." The topic shift effect results from the fact that the "similar" pairs of topics in Table 5.2 are not identical. In other words, documents which satisfy the English topic description may not satisfy the Spanish topic description, and vice versa. If the Spanish topics are viewed as "correct," this topic shift effect essentially introduces an additional source of noise into the profile training step. Alternatively, the English topics can be viewed as "correct" and the noise interpreted as having been introduced into the evaluation process. From either perspective, the result of the topic shift effect is to reduce the recall and precision measurements below that which would be achieved if noise-free relevance judgements were available. We have designed and conducted additional experiments to characterize the results of the topic shift effect as well.

The topic shift and domain shift effects are by no means the only sources of measurement errors, however. The other sources are inherent to the use of TREC relevance judgements. The principal limitation, shared by all approaches which are based on recall and precision computations, is that relevance judgements based on a topic description are subject to the assessor's understanding of the topic statement, of the document, and of the relationship between the two. Such judgements have been shown to vary significantly across assessors, with better "inter-rater reliability" demonstrated by assessors who are expert in the knowledge domain of the topic. Inter-rater reliability measurements rarely exceed 0.8, meaning that at best 80% of the documents that are judged to be rel-

evant by one assessor would be judged to be relevant by another, or even by the same assessor at another time (c.f., [131]). Thus, a system achieving a precision of 0.8 with respect to a single judge's relevance assessments (TREC relevance judgements are made by a single judge for each topic) would be operating at or near the limit of our ability to reliably measure performance.

A second limitation applies to experiments such as ours in which relevance judgements that were made during a past TREC evaluation are used. With a pooled relevance assessment methodology it is not possible to know how the documents which were not in the pool of evaluated documents would have been judged. Typically, such documents are treated as if they are not relevant when computing precision and recall. During the TREC evaluation this factor only effects precision measurements because every document selected by any system is evaluated. For evaluations performed using existing TREC relevance judgements, both recall and precision can be affected because unevaluated documents may be highly ranked. We call this source of error the "sampling effect" because it is a consequence of the sampling technique used in the TREC evaluation. The magnitude of the error introduced by the sampling effect is likely larger for the El Norte collection because fewer systems have participated in the Spanish TREC evaluation and that fact has resulted in smaller pools of assessed documents. We have designed and conducted an additional experiment which offers some insight into the impact of the sampling effect on our results.

The principal goal of our experiments is to compare the effectiveness of the three adaptive multilingual text filtering techniques that we have developed. We do this using precision and recall measurements which are subject to all four sources of error: domain shift, topic shift, inter-rater reliability and sampling.

Our results thus do not reflect the performance which an actual user might expect to achieve in a specific application. The can, however, be used to evaluate the relative effectiveness of the three techniques so long as the precision which is achieved exceeds that which would be achieved by chance. There are 57,780 documents in the El Norte collection that we use for evaluation. Table 5.3 shows the precision which would be achieved by random selection, considering the number of relevant documents that are known for each topic. The number of known non-relevant documents for each topic is also shown.

Topic	Relevant Documents	Total Documents	Chance Precision	Non-relevant Documents
SP10	206	57,780	0.004	735
SP22	346	57,780	0.005	413
SP25	359	57,780	0.006	427
SP47	77	57,780	0.001	483

Table 5.3: Precision achieved by random selection on the El Norte collection.

The experiment design in Figure 5.8 is particularly well suited to measuring the domain shift and topic shift effects. The effect of the domain shift between the UN collection and the El Norte collection can be characterized by a variation on the TT experiment. In the standardized TT experiment design, the SVD is performed on Spanish documents drawn from the UN collection. By instead performing the SVD on a sample of the the El Norte documents, the effect of the domain shift between the UN collection and the El Norte collection will be removed.[7] If an improvement in precision occurs when the domain shift effect is

_____

[7] In our primary experiments we do not want to remove the domain shift effect from the TT experiment because it is unavoidably present in the LSC experiment.

removed in this way, the magnitude of the increase can be used as a measure of the severity of the domain shift. Since the domain shift effect may be different for different topics, we have performed this "domain shift experiment" on a topic-by-topic basis. Unfortunately, we have not been able to devise a similar procedure to measure the effect of the domain shift between between the WSJ collection and either of the other two collections.

One point worth nothing is that the design of our domain shift experiment produces a perfectly fair evaluation of TT performance for some types of filtering applications. In our standard experiment design we process each document in the evaluation collection individually, accumulating the result solely for the purpose of constructing a sorted list. The "newly arrived" El Norte articles could equally well be processed as a group from start to finish in applications which do not demand real-time performance. In our "domain shift" experiments we perform the SVD on the first 1000 complete El Norte articles, a process which requires about one hour on a SPARC 20. It would certainly be possible to execute such a batch processing procedure overnight if adequate computing resources were available.

It is more difficult to characterize the topic shift effect because changing a single collection in the standard experiment design affects precision and recall measurements in three different ways. The "topic shift experiment" is again based on the standardized TT experiment, but in this case it is the profile training collection that is changed. Instead of training the profile on translated WSJ documents which are relevant to an English (WSJ) topic, the profile is trained on some of the El Norte documents which are relevant to the corresponding (Spanish) topics. This is certainly not a fair evaluation of filtering performance, but

it does remove the effect of the topic shift. The topic shift experiment actually evaluates a combination of memory and prediction, since it measures how well a system trained with one set of documents can remember those documents and also find new ones. So the performance improvement which results from training the profile using some of the El Norte documents and relevance judgements significantly overstates the effect of the topic shift.

In addition to removing the topic shift effect and the need to make predictions, the topic shift experiment also removes the effect of errors introduced by the machine translation step in the TT evaluation because the resulting evaluation is completely monolingual. A simple variation of the standard LSC experiment can be used to reveal this "machine translation effect." In the standard LSC experiment the English language WSJ documents are used for profile training. Of course, since the LSC SVD is performed on bilingual documents, documents in either language can be used for profile training. If the WSJ documents are first translated into Spanish and then used to train the profile, the resulting performance decrease will be entirely attributable to the errors induced during machine translation process. In all of our experiments we have configured the Logos machine translation system to leave unchanged any words which are not recognized. Thus our "translated WSJ documents" actually contain a mixture of Spanish and English words. This effect causes our experiment to measure the adverse effects of machine translation to understate those effects somewhat, but the experiment does provide some useful insight.

The effect of machine translation errors is interesting in its own right, since it offers some insight into the effect which further enhancements to machine translation could have on the performance of the TT technique, but it also

allows us to adjust the results we observe in the topic shift experiment. Once the "translation error effect" has been characterized, an upper bound on the magnitude of the topic shift effect can be approximated by adjusting the result of the topic shift experiment using the result of the experiment to determine the machine translation effect. In the next section we combine the observed percentage differences to produce an upper bound on the magnitude of the topic shift effect, although we do not intend to imply that the results are significant to the two figures that we typically show. It is not clear that percent differences are the appropriate measure in this case, but their use offers one way of visualizing how the two effects could be considered simultaneously. Table 5.4 summarizes the types of errors we have described.

Error Type	Source
Domain Shift	Different coverage of UN, WSJ, and El Norte
Topic Shift	Differences between Spanish and English topics
Translation	Incorrect word choice during machine translation
Inter-rater	Different interpretations of topic specifications
Sampling	Relevance assessment only available for some docs

Table 5.4: Sources of experimental error.

5.4   Results

Table 5.5 summarizes the results of the three standard experiments. LSC and TT appear to achieve comparable performance on these topic pairs, but the performance of VT is noticeably worse. While it is not possible to test this hypothesis statistically with only four topics, the relative performance of the three techniques is fairly consistent. The other obvious conclusion is that something

is wrong with the SP10/022 topic pair. Our additional experiments reveal that the problem with that topic pair is a consequence of the topic shift effect. The other topics pairs provide insight into more effects simultaneously, however, so we begin our detailed examination of the results with the best performing topic pair, SP22/008.

Topic Pair	Technique		
	LSC	TT	VT
SP10/022	0.01	0.02	0.00
SP22/008	0.17	0.17	0.12
SP25/128	0.08	0.10	0.03
SP47/123	0.07	0.06	0.01
Average	0.08	0.09	0.04

Table 5.5: Standard experiment results (precision at 0.1 recall).

Table 5.6 presents the detailed results for the SP22/008 topic pair which addresses issues related to economic projections. The results of the three standard experiments shown in Table 5.5 are underlined. Since there are 346 known relevant documents for topic SP22, a recall of 0.1 is achieved after 35 relevant documents have been found. So about one in six of the documents in the top 205 documents are relevant. This compares favorably with both the chance precision for this topic pair (from Table 5.3) of 0.005 and with the observed lower bound on performance of 0.06. This "lower bound" is measured with a variation on the TT experiment in which the untranslated WSJ articles for profile training. Only English words which also appear in both the El Norte collection and the Spanish UN documents can possibly contribute to the "lower bound" ranking. Viewed in this light, VT increases performance from one known relevant document in every seventeen to one known relevant document in eight, and LSC and TT produce

a further increase to one known relevant document in six.

LSI Training Collection	Profile Training			Bounds	
	LSC	TT	VT	Lower	Upper
Bilingual UN	<u>0.17</u>	0.14			
Spanish UN		<u>0.17</u>	<u>0.12</u>	0.06	0.46
El Norte		0.28			0.64

Table 5.6: Precision for the SP22/008 topic pair at 0.1 recall.

The results of the domain shift experiment can be seen by comparing the bottom rows in the second column. The underlined entry (0.17) is the standard TT experiment result, and the value below it (0.28) is produced by removing the effect of the domain shift between the UN collection and the El Norte collection. So the TT technique can actually produce ranked lists for this topic pair in which one in four of the top ranked documents are known to be relevant and, if a bilingual training collection that did not incur a significant domain shift were available, LSC should be able to achieve similar performance.

The effect of machine translation errors can be seen by comparing the first two entries in the top row. The underlined entry (0.17) is the standard LSC experiment result. The value to the right of it (0.14) is produced by adding the translation error effect. Recall, however, that this experiment understates the translation error effect because untranslated words which appear in the translated WSJ documents that are used for profile training are exploited by the LSC technique, but they (for the most part) rejected automatically by the TT technique.

With this result in mind, the results of the topic shift experiment can now be seen by comparing the TT experiment result (0.17) with the upper bound in

201

that same row (0.46). This "upper bound" is the precision that is achieved when training is performed using the El Norte collection (the same collection on which the evaluation was performed). As we mentioned above, although this removes the effect of the topic shift, it also eliminates the need to make predictions. The translation error effect is also removed, but that effect can at least be estimated as the difference between 0.17 and 0.14.

Overall, this topic pair shows the smallest performance degradation from the domain and topic shifts, but even in this case the impact is severe. For this topic pair, the domain shift reduces precision by 39% (0.17/0.28) and the topic shift also appears to result in at most a 45% reduction in precision (0.17/0.46, corrected by 0.17/0.14) (although this last comparison also includes the effect of substituting memory for prediction). The value in the lower right corner of Table 5.6 shows the precision achieved when the El Norte collection is used for the SVD, profile training and evaluation. This "monolingual memory test" reveals only that the LSI-mean technique is able to construct a representation of topic SP22 which will select the correct document two times out of three near the top of the list.

Table 5.7 presents the detailed results for the SP47/123 topic pair which addresses issues related to cancer research. The results here follow essentially the same pattern as those for the SP22/008 topic pair, but some important differences can be observed. The most important difference is shown in Table 5.3: only 77 relevant documents have been identified for topic SP47, while 346 relevant documents are known for topic SP22. This is a consequence of the way in which the TREC relevance assessments have been collected, and it probably does not indicate that there are actually that few relevant documents in the collection.

The first 25 Spanish topics were used in two consecutive TREC evaluations, but the second set of 25 Spanish topics have only been used in one TREC evaluation. Of the four Spanish topics we use, only topic SP47 suffers from this smaller pool of documents on which which relevance assessments have been made.

LSI Training	Profile Training			Bounds	
Collection	LSC	TT	VT	Lower	Upper
Bilingual UN	0.07	0.02			
Spanish UN		0.06	0.01	0.00	0.45
El Norte		0.17			0.47

Table 5.7: Precision for the SP47/123 topic pair at 0.1 recall.

In an effort to characterize the effect of this smaller pool, we repeated some of our experiments on the SP22/008 topic pair using only the relevance judgements which were available after only one TREC evaluation. At that time, 270 relevant documents were known (76 fewer than were known after both TREC evaluations). The results shown in Table 5.8 indicate that the reduction in precision which results from using this smaller pool of relevance judgements is somewhat smaller than the differences observed between Tables 5.6 and 5.7.

LSI Training	Profile Training			Bounds	
Collection	LSC	TT	VT	Lower	Upper
Bilingual UN	0.12	0.11			
Spanish UN		0.14	0.08	0.04	0.38

Table 5.8: Results for the SP22/008 topic pair with earlier relevance judgements.

Both the chance performance that would be achieved by random selection (shown in Table 5.3) and the observed lower bound on precision for the SP47/123

topic pair are very close to zero, a consequence of the fact that only 77 relevant documents are known. The absolute value of the domain shift is no larger in this case (0.17-0.06) than for the SP22/008 topic pair, although the percentage difference is much greater (a 65% reduction in precision). The translation error shows the same effect, with a nearly identical absolute difference (0.07-0.02) but a considerably larger percentage difference (a 71% reduction in precision).

The topic shift measurement for the SP47/123 topic pair are, unfortunately, not comparable to the measurement for the SP22/008 topic pair. Since the SP47 topic was used in only one TREC evaluation, the upper bound on precision (0.45) was computed by training on exactly the same set of documents that were being used for evaluation. So the difference between that value and the TT result (0.06) is likely increased by the fact that this is strictly a "memory" evaluation.

Whether the differences result from the smaller collection, a larger (percentage) domain shift, or a larger topic shift, the results for the SP47/123 topic pair are considerably below those obtained from the SP22/008 topic pair. But since our principal interest is in the relative performance of the three techniques, this difference is of little consequence. What is important is that the performance of the LSC and TT techniques is again comparable, that again TT achieves better performance when the domain shift effect is eliminated, and that the performance of the VT technique is worse than that of the other two.

The results for VT on this topic pair are actually quite disappointing, increasing the density of relevant documents from one in a thousand (which would be achieved by chance) to about one in a hundred. Shen constructed eight translation matrices using a different threshold on the cumulative confidence value for each. We selected the translation matrix which resulted in the highest average

precision (at a recall of 0.1) on the SP22/008 topic pair. This threshold produced an average of 4.7 translations for each of 3,931 English terms. Since 9,002 unique Spanish terms were identified as translation targets, the average number of English terms translating to a single Spanish term was 2.1. This is the factor which permits the consensus translation effect to develop. Table 5.9 shows these parameters for the best three of the eight threshold values we tried.[8] From this we conclude that an expansion factor of about 4 and a contraction factor of about 2 will maximize the performance of VT on the SP22/008 topic pair.

Matrix	SP22/008 Precision	English Terms	Expand Factor	Nonzero Entries	Contract Factor	Spanish Terms
1	0.05	3369	2.9	9648	1.3	7395
2	0.08	3931	4.7	18526	2.1	9002
3	0.06	4287	5.5	23479	2.5	9244

Table 5.9: Term alignment statistics.

In view of the disappointing results of VT on the SP47/123 topic pair, we repeated the VT experiment on that topic pair with all eight of the translation matrices that resulted from the different threshold values. No translation matrix resulted in an average precision (at a recall value of 0.1) exceeding 0.01. From this we conclude that VT performs poorly on the SP47/123 topic pair.

Table 5.10 presents the detailed results for the SP25/128 topic pair which addresses issues related to privatization of state assets. The most surprising figure in that table is the exceptionally low upper bound on precision obtained

---

[8]The "precision" shown in Table 5.9 was measured at a recall value of 0.1. It was computed using only the relevant documents known after the first TREC evaluation of topic SP22, so it should be compared with the results in Table 5.8 rather than Table 5.6.

when the El Norte collection is used for all three steps in the experiment (0.18). This suggests that for topic SP25 either the relevant and nonrelevant documents are not well separated in the k-dimensional LSI subspace, or that the pooled relevance methodology has failed to identify many of the relevant documents. The other unusual effect that can be observed in Table 5.10 is that the translation error effect is reversed for this topic pair. In other words, training the LSC profile with translated Spanish documents actually improves performance when compared to the result when the LSC profile is trained with documents in English. This suggests that LSC has failed to identify the cross-language mappings of the terms which are important in documents about privatization. The unusual translation error effect could also be an artifact of the same problem that limited monolingual performance, although the results of the standard experiments (LSC, VT and TT) are reasonably consistent with the other two topic pairs that we have considered so far.

LSI Training	Profile Training			Bounds	
Collection	LSC	TT	VT	Lower	Upper
Bilingual UN	0.08	0.13			
Spanish UN		0.10	0.03	0.02	0.10
El Norte		0.10			0.18

Table 5.10: Precision for the SP25/128 topic pair at 0.1 recall.

Table 5.11 presents the detailed results for the SP10/022 topic pair which addresses issues related to narcotics trafficking. As Table 5.3 shows, all of the values (except the upper bounds) are extremely close to that which could be achieved by random selection (0.004). This appears to result from an exceptionally large topic shift, since the performance when El Norte is used for every step

206

(0.79) approximates the best practical inter-rater reliability that we described above.

LSI Training Collection	Profile Training			Bounds	
	LSC	TT	VT	Lower	Upper
Bilingual UN	0.01	0.01			
Spanish UN		0.02	0.00	0.01	0.20
El Norte		0.02			0.79

Table 5.11: Precision for the SP10/022 topic pair at 0.1 recall.

A close inspection of the topic descriptions in Appendix B does suggest a possible explanation for this mismatch. Topic SP10 specifically requires that "The document should indicate methods used by narcotraffickers to utilize Mexico as a transit country for getting drugs into the U.S. It should include specific examples and locations and measures for stopping this activity." Topic 022, on the other hand, requires that "To be relevant, a document will report measures taken by the U.S. Government either to curb production of drugs, to curb entry into the U.S. of drugs, or to prosecute those involved in drug trafficking, laundering of drug money, or racketeering related to drugs." As might be expected, many of the relevant documents in the Wall Street Journal collection discuss prosecutions. In the topic SP22, however, the emphasis is clearly on prevention rather than prosecution. Thus, the problem results in large part from a domain shift between the Wall Street Journal and El Norte, something which we are unable to measure.

This problem also points up a deficiency of the Venn diagrams in Figure 5.9. There is, in fact, a considerable overlap between topic SP10 and topic 022, but that overlap must be interpreted in the context of the two collections of

documents against which those topics have been evaluated. In Appendix B we make explicit use of this fact to allow us to overlook the restriction on the Spanish topics to Mexico (since a large number of the documents in the El Norte collection focus on Mexico). Our inability to obtain useful results from the SP10/022 topic pair demonstrates that this affect can reduce the quality of an apparently good match as well.

Despite the limitations imposed by the topic and domain shifts inherent in our collections and by our choice to use only a single filtering technique (LSI-mean), these results clearly demonstrate that cross-language text filtering can achieve effectiveness levels that would be useful in some applications. Even with the limitations imposed by our experiment design, near the top of the ranked list the user need only examine a handful of documents for every relevant document that is found. In the next section we describe the implications of our work for future research on adaptive multilingual text retrieval.

5.5 Implications for Future Work

The test collections we have used in these experiments have imposed several limitations, the most severe of which is that only three suitable topic pairs are available. It is well known that the performance of text retrieval systems varies markedly from query to query (c.f. [129]). Although the learning behavior of adaptive text filtering systems is intended to remove one source of this variation (the user's need to express their visceral information need as a query), other sources of potential variations remain (e.g., the simple vector representation). Larger numbers of queries are thus desirable so that both the typical performance

and the degree of variation can be observed.

Topic shift appears to have reduced precision significantly in our experiments. While this has not prevented us from making general comparisons across the three techniques, small variations in performance are difficult to see under such conditions. As a result, we have not been able to reliably distinguish between the performance of the TT and LSC techniques when both are run under the same experimental conditions.

The principal obstacle to the construction of better test collections is that suitably prepared parallel bilingual and multilingual document collections are more difficult to construct than monolingual document collections, and so there is a built-in disincentive to use them in evaluations such as TREC in which the principal focus is on monolingual performance. Acquisition, alignment and (when required) the addition of Standard Generalized Markup Language (SGML) codes all require additional effort, both because bilingual and multilingual document collections are less common than monolingual collections and because they are correspondingly larger.

Since the LSC technique requires a parallel document collection and the VT technique can exploit it as well, the development of automated tools for alignment of existing bilingual collections would be of significant value. Presently such collections are typically aligned at the document level using heuristic techniques that are specific to a particular collection. The United Nations document numbers that were used by the Linguistic Data Consortium to align the UN collection are one example of that approach. Bilingual text retrieval techniques could conceivably also be used to suggest possible alignments, significantly reducing the workload of aligning less well marked collections. Ultimately, the

broad applicability of the LSC technique depends on the development of parallel document collections for domains appropriate to the documents being filtered, so the development of fully automatic document alignment techniques will be important.

Another research direction suggested by our experiments is seeding the development of the VT translation matrix with an existing bilingual term list. Although VT consistently performs at a level below that achieved by LSC and TT, our present implementation of VT is strictly corpus-based. It is possible that the performance of VT could exceed that of both LSC and our present implementation of TT if spurious translations were eliminated from the translation matrix.

Additional work on the TT technique may also be useful, although it appears from our results that the translation error effect is relatively small. Since the presence of English words in the translated documents has limited our ability to measure the translation error effect, the actual effect may be considerably larger. When better test collections become available we plan to rerun the translation error experiment without the English words in order to measure the effect more accurately. If those results indicate that the effect is sufficiently large, it would be worth investigating the development of more closely integrated machine translation systems in which the intermediate representations could be used directly by an adaptive text filtering system in a way that preserves information about unresolvable ambiguity. Such an approach would have beneficial effects on efficiency as well, since some unnecessary language generation components could be eliminated.

5.6 Summary

It is clear from our results that effective adaptive multilingual text filtering techniques can be constructed. Although the LSC and TT techniques achieve similar performance when the TT experiment is run using the Spanish UN documents, substantial improvement results when LSI is trained on the El Norte collection. Thus, at present the most effective known technique for adaptive multilingual text filtering is Text Translation. Furthermore, although the requirement to translate documents into a single language imposes a unique burden on the Text Translation technique, the existence of the translations may be helpful to the user.

A strength of the LSC technique, on the other hand, is that LSC can exploit existing bilingual documents if no suitable machine translation system is available. But the primary advantage of LSC is that it can achieve dramatically better throughput. The SVD can be computed in an hour or two, and one page documents can be rank ordered at the rate of about 1,000 per minute on a SPARC 20. Although the TT technique achieves about the same throughput on documents in the primary language, documents in other languages are translated at a rate of about 500 pages per hour. So TT and LSC demonstrate a classic tradeoff between effectiveness and computational complexity. The dramatic computational advantage of LSC should create a strong market niche for that technique.

We have shown that the Vector Translation technique also performs reasonably well on some topic pairs, although it does not achieve the effectiveness achieved by either of the other two techniques. It is the potential of Vector Translation to combine the best of the knowledge-based and corpus-based tech-

niques that makes it particularly interesting, and the performance that it has demonstrated in these experiments with only corpus-based information indicates that Vector Translation may eventually be able to achieve levels of effectiveness which rival the other two techniques.

Each of the three techniques that we have developed uses a different approach to map information across languages, and each is appropriate for different types of applications. Our principal contribution has been to show that effective adaptive multilingual text filtering techniques can be found, and that their performance can be evaluated objectively using presently available test collections.

Chapter 6

Conclusions

Adaptive multilingual text filtering is a new research area, but adaptive multilingual text filtering techniques can be based on an extensive heritage of research in related fields. Earlier work on adaptive text filtering and multilingual text retrieval provides a rich array of techniques, only some of which have been investigated in this dissertation. We have made three major contributions in this area:

- Developed and compared three techniques for adaptive multilingual text filtering using an evaluation methodology which exploits existing test collections:

    - Text Translation

    - Latent Semantic Coindexing

    - Vector Translation

- Identified fundamental limitations on the effectiveness of vector space adaptive text filtering techniques.

- Presented comprehensive reviews of present practice in two closely related areas:

  - Adaptive text filtering

  - Multilingual text retrieval

As we have seen, the adaptive multilingual text filtering problem imposes two unique demands on the design and evaluation of suitable techniques:

1. The profile training technique must develop a single profile for each information need, regardless of the languages of the training documents that are used to construct the profile or the languages of the documents that the profile is used to select.

2. The test collections used to evaluate selection effectiveness must contain sufficiently large numbers of documents in more than one language to permit training and evaluation sets to be constructed and sufficiently many topics for which relevance assessments are available to minimize the dependence of the selection effectiveness measures on the characteristics of individual topics.

The first of these demands, cross-language training, provides a somewhat richer basis for the design of suitable techniques than does the cross-language selection requirement inherent in the multilingual text retrieval problem. The Vector Translation technique is designed to exploit that potential, but it represents only the first step in exploring this issue.

The second demand, construction of suitable test collections, is a far more immediate problem. As we have shown with Text Translation and Latent Semantic

Coindexing, some techniques that were designed originally for multilingual text retrieval can be productively applied to adaptive multilingual text filtering as well. But without a suitable test collection, it is difficult to determine what combination of multilingual retrieval and adaptive text filtering techniques will result in the best performance. Fortunately, the present methodology for experimental evaluation of text filtering systems is quite compatible with this requirement, a point which we explain in greater detail below. But before elaborating on these and other issues requiring future work, we first must identify the limitations of our investigation of this topic.

6.1 Limitations

Any experimental investigation is necessarily limited, and this is particularly true of a topic as broad as adaptive multilingual text filtering. The limitations of our investigation are summarized below.

- The surveys of present practice in adaptive text filtering and multilingual text retrieval are generally restricted to material published in the English language. This limitation is particularly significant with respect to thesaurus-based multilingual text retrieval techniques, a topic of considerable interest to researchers from many linguistic backgrounds.

- The experiments which evaluated enhancements to the LSI-mean technique that were reported in Chapter 3 investigated the performance of the Gaussian User Model only on the small number of training documents that would typically be available during the initial use of an adaptive text filtering system. Furthermore, only a single procedure, regularization, was

used to compensate for the limited amount of training data.

- The Gaussian User Model experiments were designed to explore only the potential of techniques based solely on positive training examples. The work of Schütze, *et al.* suggests that profiles similar to those used in the Gaussian User Model can be useful when both positive and negative training examples are available [132].

- More generally, the Gaussian User Model explored only the potential of statistical information need models. Although Dumais has demonstrated good performance for the LSI-mean technique, clustering approaches based on set inclusion and fuzzy set theory offer practical alternatives to such statistical approaches.

- The experimental design in Chapter 5 builds upon a single adaptive text filtering technique. Although the LSI-mean technique is well suited to revealing the relative effectiveness of the three cross-language training techniques that are compared in that chapter, other known adaptive text filtering techniques may be better matched to the characteristics of individual cross-language training techniques. Furthermore, new adaptive text filtering techniques may be found which are better suited to multilingual application than any presently known technique.

- Only collections in which each document uniformly contains one of two languages have been investigated in these experiments. In practical applications, individual documents may contain more than one language, and more than two languages appear in the collection. Furthermore, we have

implicitly assumed that the language of each document is known with certainty.

- The languages used in the experiments reported in Chapter 5, English and Spanish, are more similar than many language combinations for which practical applications might be envisioned. Important issues such as word and sentence identification arise in some languages, and problems of word-to-phrase translation and compound nominal formation are much more significant in more dissimilar language pairs.

- Except for the variations described in this dissertation, the parameters for SVD computations, LSI dimensionality reduction, and the SMART experimental text retrieval system were fixed for each set of experiments. Typically these parameter choices were motivated by the reported results of other researchers, although a few of the parameters were experimentally optimized for the specific test collection and the experiment design used in this experiment. Furthermore, complete documents were used for the profile training and evaluation steps regardless of their length and the number of topics they addressed. This approach is suitable for determining the relative performance of the three cross-language training techniques, but it does not reveal the optimum performance which could be achieved by any single technique.

- The multilingual text retrieval techniques that were applied in Chapter 5 (Text Translation and Latent Semantic Coindexing) are only two of many known techniques which could be used as a basis for adaptive multilingual text filtering. Controlled vocabulary thesaurus-based techniques could

clearly be applied in limited domains, and even the simple cross-language relevance feedback technique proposed by Fluhr could easily be investigated with the same experiment design.

- Only a single machine translation system was used in the Text Translation experiment, and the performance of that system was not manually optimized for either the domain of the test collections or the topics being used. Thus, the effects of manual addition of domain knowledge (for which the Logos system provides extensive capability) or of substituting a less sophisticated system (perhaps with less broad coverage) were not evaluated. Because the Logos machine translation system was used in unmodified form, performance improvements which could result from exploitation of intermediate representations in which unresolvable ambiguity is preserved were not investigated.

- Only a single term alignment technique was applied in the Vector Translation experiment. Furthermore, the word alignment software used in that experiment was originally intended for use with human postprocessing to produce high-quality term lists. The relatively low false alarm rate desired in that application may not provide the best possible basis for the consensus translation effect.

- The severity of the topic and domain shift effects and the small number of sufficiently similar topic pairs limit the ability to generalize the relative performance results reported in Chapter 5 to other topics or test collections.

- The experiments reported in Chapters 3 and 5 evaluate only selection effectiveness and, to a lesser extent, efficiency. While these are fundamental issues that must be addressed before this technology can be applied to solve practical problems, usability issues will be equally important to the construction of practical applications. Furthermore, the inter-rater and intra-rater variations in normative relevance judgements limits the fidelity with which even the effectiveness measurements can reflect the value which an individual user would place on the results of a specific interaction with an adaptive multilingual text filtering system.

6.2   Future Work

The principal contribution of this dissertation is our demonstration that adaptive multilingual text filtering is both possible and practical. Although existing text collections limit our ability to evaluate the effectiveness of alternative techniques, they do permit us to draw the clear conclusion that the levels of effectiveness which can be achieved will be sufficient to justify application of this technology to a number of important problems. In this section, we identify three critical research issues which, together with this dissertation, would establish the foundation for further work in this area. We follow that with some suggestions for additional research topics which could productively be investigated once the necessary foundation has been laid.

By far the most important issue which must be resolved is the construction of suitable test collections. Of the two techniques suggested in Chapter 5, assignment of relevance judgements to an existing parallel multilingual document

collection is by far the more practical approach. Large collections of translated documents exist already, and some collections contain more than two languages. Because such documents could also be used for monolingual text filtering evaluations and both monolingual and multilingual text retrieval evaluations, their development should be a matter of widespread interest. The Text REtrieval Conference, with a pooled relevance assessment methodology and a large and diverse set of participating systems, presently provides the best venue for such an approach.

The principal difficulty to be overcome before relevance judgements can be assigned to an existing collection is the negotiation of rights to use the collection for the evaluations which are contemplated. Since the level of preprocessing required to prepare a parallel multilingual document collection for research use is significantly greater than that required for a monolingual document collection, negotiation of rights to use an existing research collection or construction of a new parallel collection may require that additional resources be devoted to the multilingual aspects of the TREC evaluation. The results in Chapter 5 clearly demonstrate that such an investment would be productive, and furthermore that it is required if we are to obtain an accurate assessment of the potential offered by alternative techniques for adaptive multilingual text filtering.

Simply constructing suitable test collections would suffice to enable investigation of many of the additional research topics that we identify below. Two additional steps must be taken, however, before any practical applications can be implemented with confidence. The most basic of these is to investigate the application of other adaptive text filtering techniques with the available cross-language training techniques. The LSI-mean technique was an attractive choice

for our experiments because it was particularly well suited to one cross-language technique (LSC), and it could be applied easily to the other two techniques as well. Once suitable test collections are available, our experiment design should provide adequate insight into which cross-language training technique works best with the LSI-mean technique. But before developing a large-scale application, it would be helpful to identify which combination of cross-language training and adaptive text filtering techniques provides the best performance.

To accomplish this goal, slightly different approaches will be needed for each of the three cross-language training techniques. The most general cross-language training technique is Text Translation, since Text Translation reduces the multilingual problem to its monolingual equivalent. An experiment comparing LSI-mean with relevance feedback in a probabilistic text retrieval system such as Inquiry and with a linear neural network implementation of logistic regression such as that used by Schütze, *et al.* would be a good first step. The scope of a Vector Translation experiment would be limited to techniques such as LSI-mean and the inference networks used in Inquiry which build and manipulate representations that can be expressed as vectors. Shuütze's linear neural network technique may prove to be less well suited to Vector Translation because the "spreading" effect shown in Figure 5.3 may preclude accurate identification of the most informative terms. Latent Semantic Coindexing offers an even more restricted choice of approaches, but Hull's "local LSI" technique should certainly be investigated [68]. Berry and Young's results also suggest that adjusting the size of the aligned passages that are extracted from the training set can significantly affect the performance of Latent Semantic Coindexing [7].

This second step will provide a sound basis for choosing a combination of

multilingual training and adaptive text filtering techniques, but it would be wise to investigate usability issues as well before attempting to apply these ideas in a large-scale application. Some usability issues, such as whether users find a single ranked list more or less useful than separate ranked lists for each language, will be applicable to a large range of applications. Others will be more specific to a desired application. For example, making translated text available on demand might be very useful to members of an electronic discussion group in which ten languages are used, but considerably less important for the customers of a newswire filtering service providing articles in two languages to a predominantly bilingual subscriber base. Thus, usability evaluation is probably best viewed as a prototyping process, and the results of the prototype evaluation will thus be most useful when it is developed for a domain which matches the intended application as closely as possible.

Because interactive usability assessment techniques can be applied without large test collections for which relevance judgements are available, it is tempting to consider skipping the first two steps and proceeding immediately to develop a prototype of a practical system. The results in Chapter 5 demonstrate clearly that such an approach could be made to work, at least for interactive applications which require only moderate precision at low values of recall. Unfortunately, interactive evaluations on dynamic document streams reveal little insight into whether any observed limitations result from features of the user interface or the effectiveness of the algorithms which have been implemented. Furthermore, uncharacterized changes in individual preferences and in the content of a dynamic document stream make it difficult enough to even recognize the effect of changes to the user interface. Experimental evaluation of filtering effectiveness certainly

does not address all of the questions which must be answered. But because it does provide a sound basis for making some of the necessary design decisions, it is a logical prerequisite to usability assessment.

Adaptive multilingual text filtering raises some deeper research issues as well. As we observed in Chapter 4, lessons learned in a monolingual context about the importance of issues such as word sense disambiguation and phrase indexing may not always be directly transferable to multilingual applications. It should thus not be surprising if it turns out that the better approaches can be found to adaptive multilingual text filtering than the simple combination of an adaptive text filtering technique with a technique inspired by multilingual text retrieval practice. Vector Translation represents one such approach, but it by no means exhausts the potential design space of such techniques.

With our present state of knowledge it is not even possible to state conclusively that the performance of adaptive multilingual text filtering techniques will remain bounded by their monolingual counterparts. As Vector Translation's consensus translation effect demonstrates, although the multilingual aspect of the adaptive multilingual text filtering problem introduces additional challenges (e.g., translation in the face of polysemy), it brings along additional sources of information which can be exploited as well. It seems possible that in some applications this additional information could more than compensate for the additional problems. Thus, research devoted to developing techniques which exploit unique characteristics of the adaptive multilingual text filtering problem could prove quite worthwhile.

A number of less sweeping, but perhaps no less important, research topics are suggested by the limitations of this dissertation which were identified above.

Evaluation of the Gaussian User Model on a test collection for which more training documents are available is an obvious step, and the author is pursuing that during the TREC-5 evaluation. If those experiments demonstrate an improvement over the effectiveness of the LSI-mean technique, future investigations could explore alternatives to regularization that might improve performance during the early stages of interactive use.

The other research direction suggested by our discussion of the Gaussian User Model results is to exploit the information provided by the negative as well as the positive training examples. When explicit feedback is available, there are typically far more negative than positive training examples, so this source of information could be very informative.

Several straightforward extensions to the multilingual text filtering experiments reported in Chapter 5 could also provide valuable insights. The most obvious extension is the introduction of additional languages. If TREC relevance assessments were to become available for the UN collection, it would be possible to repeat the experiment for English, Spanish and French (or any two-language subset of the three) by simply partitioning the collection into separate language training, LSI training and evaluation sections. Parallel collections including languages in which the lack of inter-word spaces makes word recognition a challenge (e.g., Chinese) would be particularly useful in this regard, although they would likely be correspondingly more difficult to construct.

The excellent relative performance of the Text Translation technique, particularly when it is possible to train LSI on the evaluation set, suggests several profitable lines of investigation which explore the effect of different approaches to Text Translation. Fast translation is the most pressing need, because the

low throughput of the Logos machine translation system (relative to the Latent Semantic Coindexing technique) places severe constraints on the amount of text which can be translated to support practical applications. Thesaurus-based controlled vocabulary techniques offer a radical approach which might be useful in limited domains, and word-for-word or statistical translation approaches (c.f., [16]) might achieve acceptable text filtering performance with greatly improved throughput.

A second important topic for Text Translation research is the potential improvement in both effectiveness and efficiency which could result from using the intermediate representation of a machine translation system as a basis for adaptive multilingual text filtering. Effectiveness is potentially enhanced because unresolvable ambiguity need not result in the arbitrary choice of a single word sense, and efficiency would clearly be enhanced because many of the difficult issues of language generation such as morphology and word order choices can be eliminated entirely. Furthermore, implementation efficiencies can be achieved if a common interlingual representation is used for analysis of texts in several languages. Interlingual machine translation is still an area of active research, but application of that emerging technology to adaptive multilingual text filtering should be somewhat simpler because only the analysis component for each language need be implemented. These three considerations (efficiency, effectiveness, and ease of implementation) apply equally well to the multilingual text retrieval problem, and the EMIR project represents the present state of the art for the partial translation approach [141].

Essentially the result of using word-by-word translation to an intermediate representation, Vector Translation occupies the other end of this spectrum of

Text Translation variations. Because Vector Translation is amenable to both corpus-based and knowledge-based approaches, it offers a unique set of potential variations which are worth exploring. The most obvious potential variation, one which we discussed in Chapter 5, is to seed the term alignment technique with an existing multilingual term list that specifies the (possibly domain specific) allowable translations for each term. In that case, the corpus would be used to add information (translation probabilities) to an existing knowledge structure. The alternative would be to create a noisy knowledge structure directly from the language training corpus and then edit this structure by hand to improve the quality of the resulting mapping. Although human postprocessing is a far more labor-intensive approach, it does not depend on the existence of a multilingual term list for the language pair and domain in question. In fact, if a corpus-based approach is contemplated to generate such a term list for some other application, the additional effort to maintain and enhance the associated translation probabilities may turn out to be quite small. Thus, it would be worth investigating both techniques for forming the translation matrix, and consideration should be given to the development of postprocessing tools which use constraints that humans can more easily generate (e.g., "more common" and "less common") to improve the translation probability assignments.

Another important variation on Vector Translation would be incorporation of an effective phrase indexing technique. Hull and Grefenstette have shown that translating phrases rather than terms significantly enhances the effectiveness of an otherwise unconstrained term expansion approach to multilingual text retrieval [69]. The existence of fewer translations for the phrases would also diminish the consensus translation effect, however, so it is possible that

the beneficial effects of phrase indexing could be somewhat smaller for Vector Translation than for multilingual text retrieval.

Applications which demand higher levels of effectiveness than can be achieved by any individual adaptive multilingual text filtering technique might benefit from the development of an effective strategy for merging the output of several different techniques. Since, for example, Text Translation and Latent Semantic Coindexing exploit different sources of cross-language mapping information, the best possible performance when the two techniques are used together would likely be significantly better that the performance of either technique alone. Work by Voorhees, *et al.* on the related problem of fusing the output of monolingual text retrieval results produced by two different algorithms [157] suggests that it may be practical to achieve 90% of the precision that would result from a omniscient optimal merging strategy. And when three or more significantly different systems are available, experience from the field of optical character recognition suggests that the error rate can be reduced 65% or more [28]. Although such approaches are resource intensive, the potential for improved effectiveness could significantly expand the range of applications to which adaptive multilingual text filtering can be productively applied.

Another way to increase effectiveness would be to integrate collaborative filtering with content-based adaptive multilingual text filtering techniques like those we have described. As we observed in Chapter 2, collaborative filtering techniques are inherently independent of language, and they provide an additional source of information which can be exploited to increase the fidelity of the user model. Much remains to be done, however, before collaborative filtering can be relied upon to provide useful information. Among the open issues that

we identified in Chapter 2, the most crucial to resolve are the use of implicit feedback and the development of suitable evaluation techniques for collaborative filtering systems which are able to achieve the requisite critical mass of users. Only after those issues have been addressed will it become possible to discern the correct balance between the content-based and collaborative techniques for adaptive multilingual text filtering.

Finally, there are also a number of relatively simple variations on the experiments reported in Chapter 5 which would be worth investigating. Probably the most significant variation for practical applications would the substitution of word stems rather than morphological roots when forming terms from the words which appear in the documents. Morphological roots are helpful when integration with existing term lists is foreseen (e.g., the seeded Vector Translation technique), but significant advantages resulting from their use in other cases have yet to be demonstrated. Thus, the use of word stems rather than morphological roots could improve efficiency without adversely affecting performance in some applications.

Other variations on those experiments could reveal parameter settings which improve the effectiveness of one or more of the techniques. For example, the optimal value of k (the number of dimensions that are retained in the LSI step) may vary across techniques, so experimental determination of this value for each technique could be productive. The same potential exists for other parameters as well. For example, it is possible that the Vector Translation technique might demonstrate considerably better performance with a term weighting strategy in which the dependence on term frequency is linear rather than logarithmic.

Another useful variation would be to restrict the size of the training set

within each topic in order to determine the learning rate of the three adaptive multilingual text filtering techniques. We have generally used all of the available training data in our experiments, so our results do not reveal whether one technique would outperform the others during the early stages of interactive use. Because the variations we seek to measure may be relatively small, such an experiment would best be done after truly multilingual test collections become available.

6.3   Summary

We have demonstrated that adaptive multilingual text filtering is both possible and practical. Two of the three techniques we have developed each fill a different niche in terms of cost-benefit tradeoff. Text Translation achieves the highest effectiveness, increasing the density of relevant documents from one in fifty to nearly one in three at the top of the ranked list, but it suffers from a relatively low throughput of 10 pages per minute for documents which require translation. Latent Semantic Coindexing achieves only half this density of relevant documents, about one in six, but does so with a language-independent throughput that is two orders of magnitude better, around 1,000 pages per minute. Text Translation requires extensive knowledge engineering, while Latent Semantic Coindexing is exclusively corpus-based. Vector Translation, by contrast, can benefit from a mix of knowledge engineering and automatic analysis of parallel corpora. Although vector translation has yet to demonstrate competitive performance, we believe the technique has the potential to approach the effectiveness of Text Translation while equaling the efficiency of Latent Semantic Coindexing.

We also investigated the Gaussian User Model, a family of techniques which extend Dumais' LSI-mean monolingual text filtering technique to account for direction-specific variations that are observed in the training vectors. By demonstrating that no member of this family of techniques achieves improved performance (at least during the early stages of interactive use) we have illuminated a fundamental limitation on the performance of vector space techniques which are based solely on positive training examples.

Much still remains to be done, however. While we have shown that suitable text representations, cross-language mapping techniques, and user models are now known, we have also demonstrated that further development of adaptive multilingual text filtering systems will require a significant investment in the construction of suitable test collections. Once that key ingredient is in place it will become possible to resolve many of the remaining open issues that we have identified in this chapter.

The capacity and connectivity of the global information infrastructure is expanding rapidly, making it possible to convey enormous quantities of text on a world wide basis. Examples of the social and economic importance of exploiting this information abound, but without automated assistance the volume of available information will easily overwhelm any individual. It is possible to retrieve relatively static information from large text collections in a language-independent manner, but filtering a dynamic document stream must presently be done on a language-by-language basis. The research presented in this dissertation is the first step towards the construction of systems which can cross the language barrier for dynamic information as well.

Appendix A

SMART Modifications

This appendix describes our modifications to the SMART experimental text retrieval system. This information is not intended as a tutorial on SMART, so we assume a fairly intimate familiarity with the design and implementation on SMART in our presentation. Most actual coding details are omitted in the interest of clarity because we plan to make the code which implements the changes available.

A.1 Software Availability

The SMART experimental text retrieval system was developed at Cornell University in the 1960's [129]. The present version, version 11.0, is available for non-commercial research use by anonymous File Transfer program (FTP) from ftp://cs.cornell.edu/pub/smart. A number of standard text retrieval test collections are available at the same location, including the Cranfield collection that we used in Chapter 3. SMART is coded in Kernighan and Ritchie C and it compiles without difficulty on a SPARC 20 running SunOS 4.1.3.

SVDPACKC is a set of ANSI C routines that are designed to compute the

Singular Value Decomposition (SVD) of large sparse matrices. These C routines are based on earlier FORTRAN routines coded by Berry at the University of Illinois. SVDPACKC is available from the Netlib repository at the University of Tennessee through the World Wide Web as http://netlib.org/svdpack/svdpackc.tar.Z. The routines we use (from the files las2.c and las2.h) compute the SVD using the single vector Lanczos method.

We have modified SMART extensively for our experiments, and we have made some of our modifications available through the World Wide Web at http://www.ee.umd.edu/medlab/filter/smart. The modifications for Spanish character handling that we describe in Section A.2 are distributed as a single tar file which, when unpacked over an unmodified copy of SMART version 11.0, replace all of the routines necessary to handle documents which use the ISO 8859-1 character set. Modifications to SMART require the express permission of the SMART project at Cornell University, and that permission has been obtained for the Spanish character set modifications.

We plan to request permission to distribute the remainder of our modifications to SMART version 11.0 that are necessary to duplicate our experiments as a second tar file that can be unpacked in the same way. If distribution of those modifications is approved, they will be made available over the World Wide Web at the same location.

We are not at liberty to redistribute the test collections we have used, but they are available to members of the Linguistic Data Consortium at the University of Pennsylvania.[1] Information about the Logos Machine translation system

---

[1]Linguistic Data Consortium, 441 Williams Hall, University of Pennsylvania, Philadelphia, PA 19104-6305 USA

can be obtained from the Logos corporation,[2] and information about the soft-
ware used to generate the morphological roots we used is available from the
Rank Xerox Research Centre./footnoteRank Xerox Research Centre, Grenoble
Laboratory, 6 chemin de Maupertuis, 38240 Meylan France

A.2   Spanish Character Handling

TREC Spanish data is coded with the ISO 8859-1 character set, but SMART
is designed to process only 7-bit ASCII characters. In this section we describe
the modifications we made in order to process 8-bit characters from the ISO
8859-1 character set. The ISO 8859-1 character set is also used for Afrikaans,
Basque, Catalan, Danish, Dutch, English, Faeroese, Finnish, French, Galician,
German, Icelandic, Irish, Italian, Norwegian, Portuguese and Swedish, so these
modifications may be of interest to other researchers as well.

Out modifications are based on a brief outline of the required changes posted
to the `smart-people` mailing list by Chris Buckley on April 10, 1995. The
first step was to revise the parse table in the `src/libindexing/token_sect.c`
routine. The first 128 entries in the revised table are unchanged from the original.
The high block of control characters is not yet assigned a standard interpretation
in ISO 8859-1 so we coded them as control characters. Several other characters
could be assigned different interpretations:

- Decimal code 160 (Hex A0) is a "no break space" in ISO 8859-1. It is
  parsed as an ordinary space by our table.

- Decimal code 177 (Hex B1) is a "plus-minus sign." It is parsed by our

---

[2]Logos Corporation, 111 Howard Boulevard, Suite 214, Mount Arlington, NJ 07856 USA

table as punctuation rather than as a sign that can begin a number.

- Decimal codes 188-190 (Hex BC-BE) are fractions (one quarter, one half and three quarters), but they are parsed by our table as punctuation.

Other SMART routines use the ASCII character type macros and functions in the ctype.h standard library file rather than the `sm_ctype` definitions. Unfortunately, the standard ctype.h only handles 7 bit ASCII. The are two approaches to solving this problem. One is to reimplement ctype.h to handle ISO 8859-1 characters and the other is to replace the toupper() and tolower() functions, replace the isupper() and islower() macros with functions, eliminate all calls to the isascii() macro, and convert the arguments to these functions from char to unsigned char. The original toupper() and tolower() built-in functions leave character codes above 127 (Hex 7F) unchanged and the isupper() and islower() macros treats all character codes above 127 (Hex 7F) as neither upper no lower case case. Left uncorrected, this would prevent SMART from bringing capitalized characters in the extended set to lower case. We have chosen the second approach.

Once the new files were coded we changed every reference to isupper() and islower() in the following files. It was also necessary to change "char *" to "unsigned char *" for every parameter that is passed to one of the four new routines. The new functions were also added to the libindexing functions section of the `src/h/functions.h` file in order to compile cleanly.

In the process of modifying the sample TREC control files that are in Sample/trec to work with the original 250 MB of Spanish TREC data we found that `src/liblocal/libindexing/pp_trec.c` was not set up to handle the ART-NUM field. It also helps to have the HEADLINE (rather than only HEAD) field

defined to facilitate interactive use. Adding the following lines to the initialization data for the `pp_sec_trec[]` array in that file, SMART works fine on the Spanish TREC data:

```
"<ARTNUM>",    8, pp_discard, '-', PP_DISPINC,
"<HEADLINE>", 10, pp_copy,    't', PP_DISPINC,
```

We have not included Spanish stemming rules in our modifications because we have instead used morphological roots that were determined in a preprocessing step performed by David Hull at the Rank Xerox Research Center. An extremely simple set of Spanish stemming rules can be found in [18].

With these modifications, the Spanish stopword list from the SMART FTP site, and specification of `stem_wanted 0` to suppress application of English stemming rule we successfully indexed the original 250 MB of Spanish TREC data and run them against the first 25 Spanish queries. The average precision is reasonable (0.3004 for "nnc" documents and "ntc" queries). Interestingly, before the character mapping fixes, the average precision was 0.3006. Although the change is certainly not significant, it would have been more encouraging if it had been in the other direction.

In Spanish it is permissible to drop the accent from capitalized characters, and that convention is followed in the Spanish TREC data (and in the UN Spanish corpus as well), both for capitalized words (such as section headings and bylines) and for the initial capital at the beginning of a sentence. So the only time the ISO 8859-1 character handling actually helps is when a tilde appears over a capital N (which does happen in the TREC data and other corpora). In a language where accents are typically retained on capitalized letters, this conflation would be more useful.

235

We posted more detailed version of this description to the `smart-people` mailing list on July 5, 1996 and it is available from the archives at the SMART FTP site.

A.3   Latent Semantic Indexing

Latent Semantic Indexing (LSI) was implemented using the las2.c and las2.h files from SVDPACKC. The first step was to convert those routines from Kernighan and Ritchie C to ANSI C in order to eliminate the need for separate compilation. We made three other modifications to the las2.c file:

- Obtain array pointers and execution parameters from parameters to the las2.c routine rather than from a file. This avoided the need to write the compressed matrix to a file and then read it in again, although this comes at the cost of a considerable amount of memory when the sparse matrix is large.

- Write the output parameters (number of columns, number of rows, and number of singular values) and the singular values to the "lav2" output file that was already being produced for the singular vectors. The original output format produced that additional information only in human-readable form and did so in a separate file.

- Suppressed the production of the "lao2" output file.

A new file, which we call `libindexing/svd.c`, was created to provide an interface between the modified SVDPACKC routines and the SMART system. The svd routine in svd.c performs the following functions:

- Reads parameters from SMART parameter files for output file names (SVDPACKC's "lav2" file and the "map" file we describe below), SVD parameters (lanmax, maxprs, endl, endr, kappa), and control flags (such as "reuse a prior SVD" and "also write the sparse matrix file to disk").

- Reads the SMART sparse vectors produced during the indexing step and produces the Harwell-Boeing sparse matrix format that is required by the SVDPACKC routines. SMART stores each sparse vector as a linked list of term identifiers called "concept numbers" and their associated weights. The Harwell-Boeing sparse matrix format uses separate vectors for the row indices, column indices and values [9].

- Produces a "map" file which associates each row in the sparse term-document matrix with the associated SMART concept numbers. SMART concept numbers are hash table indices, and the map file is a dense list of concept numbers in ascending numerical order. The concept number associated with a row in the sparse term-document matrix is found at the corresponding position in the table, and a routine (`find_row`) is provided to return the row associated with any concept number.

The file `libindexing/index_coll.c` has been changed to take a parameter specifying the SVD routine to be used, and the the svd.c routines are specified by including the following lines in the SMART control file:

```
doc.store                       index.store.store_vec_aux
svd                             index.top.comp_svd
```

The first line (or the alternative value `index.store.vec`) is needed because sparse vectors are not stored by the routines selected by the default setting for

that parameter. The `svd` parameter should be included exactly as specified. A second alternative, `index.top.no_svd` was originally intended to suppress the SVD computation. This option was implemented instead with the `reuse_svd` parameter to the svd.c routines.

A second new file, which we call `libretrieve/sim_lsi.c` computes the similarity between the LSI feature vectors for a query and a document. This file was not used in any of the experiments reported in this dissertation, but it is included in the distribution for completeness. The file was specifically designed for text retrieval experiments in which the queries were appended to the document collection before the SVD was computed, so the document vectors are found using the D matrix rather than the less efficient T matrix technique described in Chapter 3 that much be used for previously unseen documents. Code for the T matrix technique can be found in the files described in Section A.5 if that functionality is needed in these simple routines. The `sim_lsi.c` routines are included by adding the following line to the SMART control file:

```
coll_sim                        retrieve.coll_sim.seq
seq_sim                         retrieve.ctype_vec.lsi
```

A.4   Gaussian User Model

The same SVD generation routines were used for the Gaussian User Model experiment, but a more complex data collection method was required to to implement the cross-validation method. The new `libretrieve/sim_gauss` file contains the routines which compute the Gaussian User Model distance measure and return it's inverse for use by SMART's ranking routines (which assume that larger val-

ues are better). Profiles are formed by finding the mean of all of the documents known to be relevant to a topic except for one document, the index of which is specified as a parameter in the control file. The `libretrieve/ret_tr_rr.c` file was modified to produce `libretrieve/ret_tr_rr.filter.c`, a version designed to identify the rank of the excluded document and print it (along with the index of the excluded document, the topic number, the value of α and the number of relevant documents). Parameters to the two files included the name of the output file for the ranks and the desired value of α (which was named "shrink" in these routines because α controls the shrinkage of the hyperellipse towards a hypersphere). The following additional lines must also be included in the control file to select the correct subroutines:

```
coll_sim                        retrieve.coll_sim.seq
seq_sim                         retrieve.ctype_vec.gauss
retrieve.output                 retrieve.output.ret_tr_rr_filter
```

Shell scripts are used to repeatedly run SMART for each desired combination of excluded document index and value of α. For the Gaussian User Model experiments reported in Chapter 3 the 600 or so iterations required consumed about 3 days on a SPARC 20. A standalone program called `trec_eval` from the file `trec2_eval.shar` on the SMART FTP site was used to compute the average precision values reported in Chapter 3 using the stored rank values for the excluded documents from each iteration. The file `trec_sort.c` contains a small program that was used to select the results for a single value of α and produce a format compatible with the input specification of the `trec_eval` routine. The basic technique used in `trec_sort.c` is to construct an arbitrary total order on the documents subject to the constraint that every excluded doc-

239

ument (there are at most 39 excluded documents for any topic in the Cranfield collection) must appear at the rank specified in the output file generated by `libretrieve/ret_tr_rr.filter.c`. In the case of a tie, the next unused rank is assigned to an excluded document. The `trec_sort.c` file was only used for the Cranfield collection, so the number of topics and documents in that collection are coded in the program rather than provided as parameters. Modification for other collections should be quite straightforward.

A.5   Adaptive Multilingual Text Filtering

The special-purpose design that was developed for the Gaussian User Model experiment proved inadequate for the adaptive multilingual text filtering experiments because the top-level experiment design in Figure 5.8 from Chapter 5 required the ability to substitute different collections for each of the three steps (LSI training, profile training, and evaluation). Each step is implemented with a separate set of control files, and separate SMART data structures are used for each step. The same SMART "dict" file, which contains the mapping from terms to concept numbers, is used in each step, but the introduction of new terms is prevented by using the "map" file that was described above without modification after the term-document matrix is built. Thus, only terms contained in the LSI training collection are used by the SMART routines in subsequent steps.

Figure A.1 shows the LSI training step for the Latent Semantic Coindexing (LSC) and Figure A.2 shows the corresponding step for Text Translation (TT) and Vector Translation (VT). Modified versions of the TT experiment in which El Norte was substituted for the UN Spanish data are implemented simply by

substituting the first 1000 El Norte documents instead of the Spanish versions of the 1992 UN documents. The `combine_morph` routine is a standalone program which combines the morphological roots from the last 400 tokens of the English UN documents with the corresponding number of morphological roots for the Spanish UN documents as described in Chapter 5. Two new SMART files were created as well. The `libconvert/matrix.c` file contains routines to write the las2.c input matrix in Harwell-Boeing form, and the `libconvert/svd.c` file contains a modified version of the `libindexing/svd.c` file which reads this data and calls the las2.c routine. These additional routines were added in order to permit the use of SMART "ltc" term weights as described in Chapter 3 and they were used in the "ltc" runs reported in that chapter as well.

Relevant Wall Street Journal articles for TREC topics 008, 022, 123, and 128 were preprocessed as shown in Figure A.3. Figure A.3 shows the profile training step for the TT and LSC experiments. Profile generation is done using a newly created +`libconvert/profile.c` file. In the VT experiment the vector translation itself is actually performed during the profile training step. That process is shown in Figure A.5. The vector translation itself is done using a new `libconvert/vt.c` file. The routine is a standalone program which is used to preselect the relevant documents from the Wall Street Journal collection. The known nonrelevant documents are also segregated by this routine, but they are not used in our experiments. The standalone routines `clean_wsj` and `logos_to_iso` remove the optional unknown word markers from the Logos machine translation system output and then convert from the Logos Spanish character set to ISO 8859-1 characters. The `clean_wsj` routine can be easily modified to reject unknown words if desired.
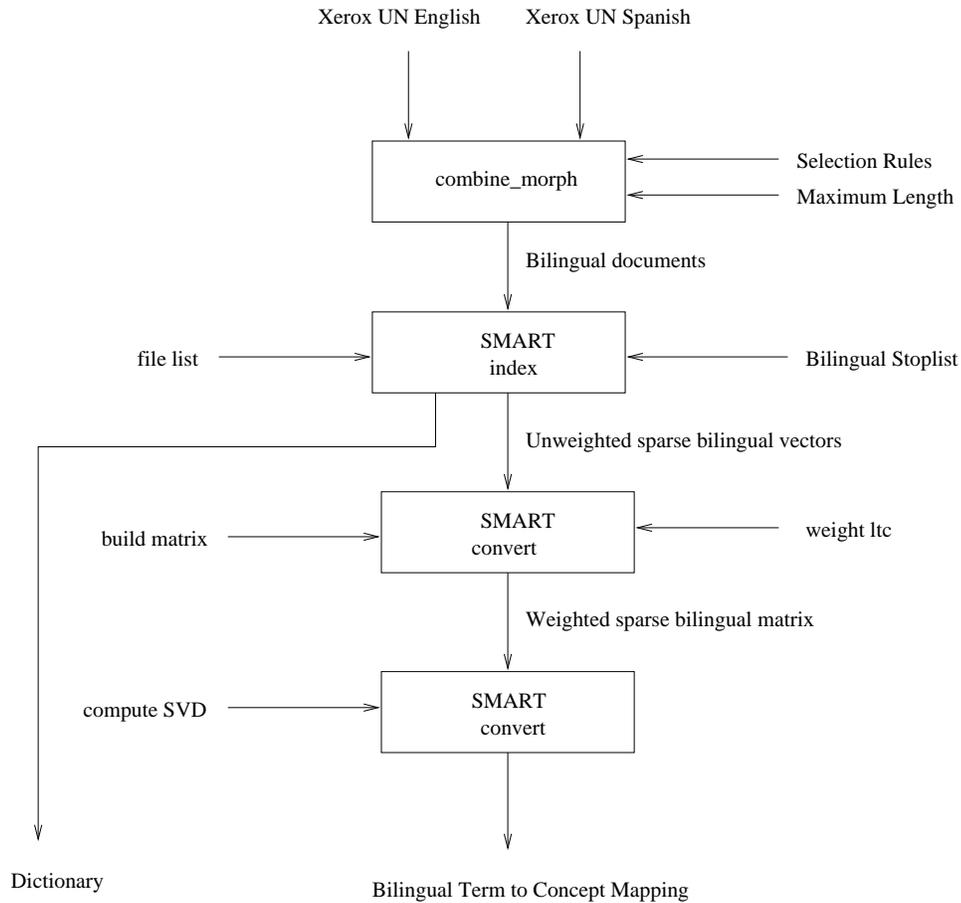
Figure A.1: LSI training step for the LSC experiment.

Figure A.6 shows the evaluation process, which is common to all three of the basic experiment as well as all of the variations that are used to evaluate topic shift, domain shift and translation error. The new `libretrieve/sim_filter.c` file contains the routines used for the retrieve block in that figure.

In these steps the generation of morphological roots by David Hull at the at Rank Xerox Research Centre is not explicitly shown. Five document sets were processed by Rank Xerox:

1. The entire UN Spanish collection

2. The entire UN English collection
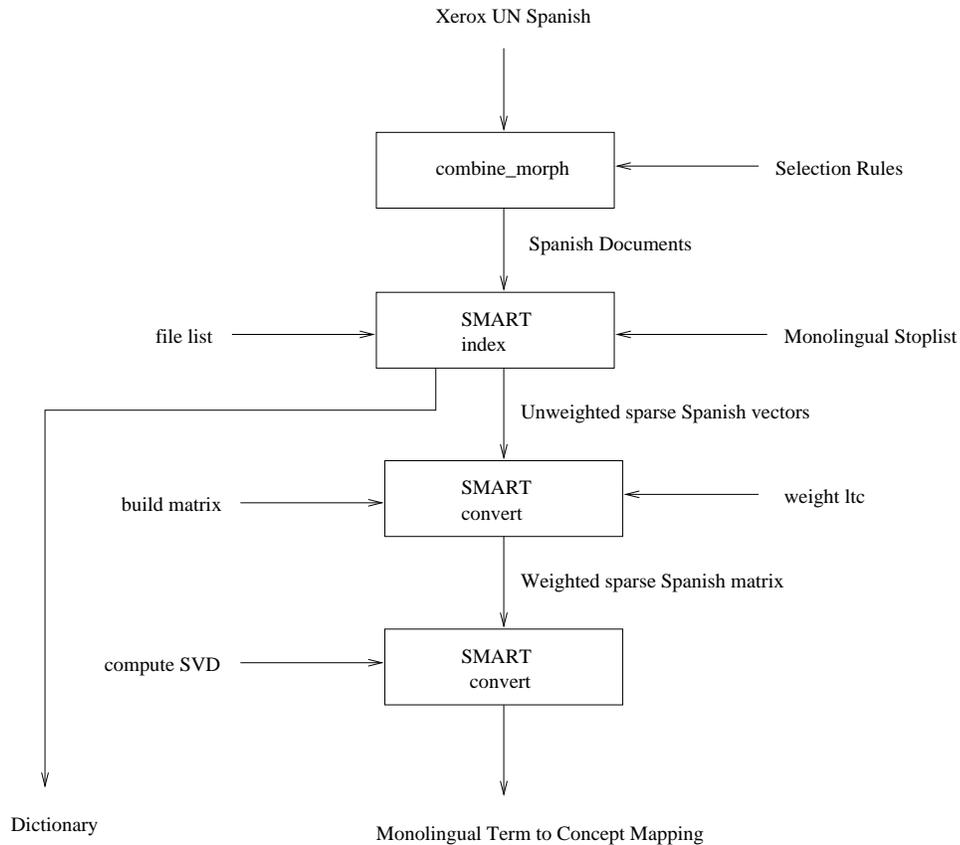
Figure A.2: LSI training step for the TT and VT experiments.

3. The entire El Norte collection

4. The relevant Wall Street Journal articles for each topic.

5. The Logos translations of the relevant Wall Street Journal articles for each topic.

In the last set, three of the relevant documents for topic 123 (articles WSJ920203-0030, WSJ920205-0103 and WSJ920311-0035) were omitted from the translated set due to difficulties coordinating the translation process with the generation of morphological roots.
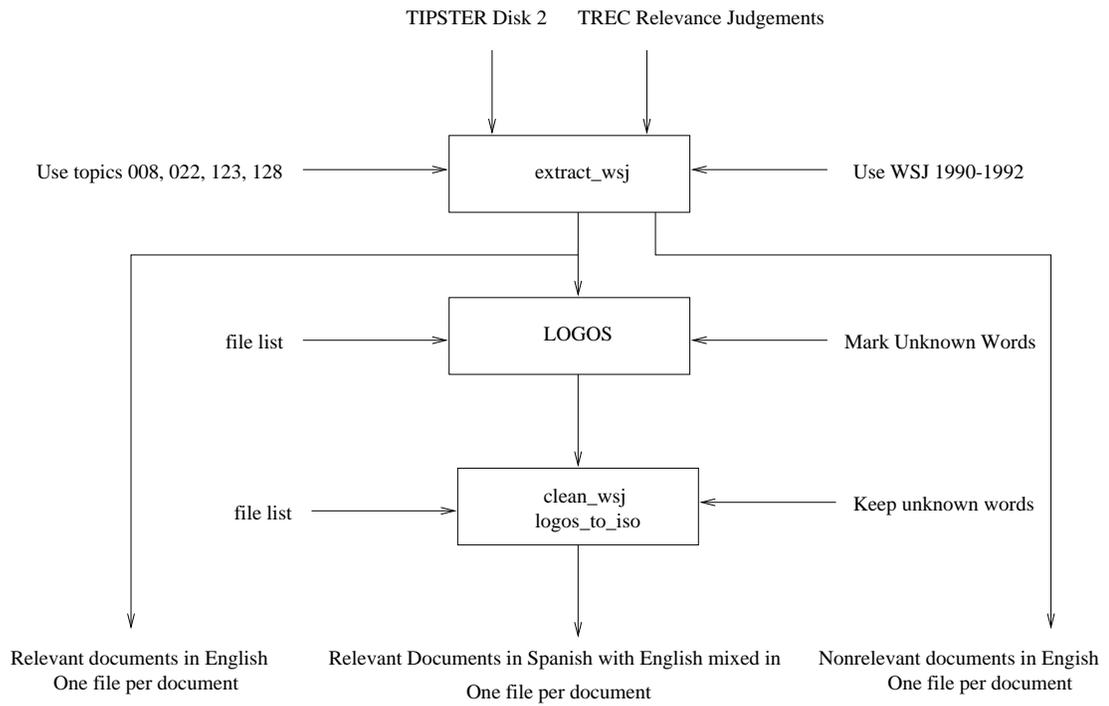
Figure A.3: Profile training data preparation.

```
                        Training Text
              Single topic, one document per file

                             │
                             ▼
              ┌──────────────────────────────┐
Dictionary ──────────▶│          SMART         │◀────────── Stoplist
              │          index         │
              └──────────────────────────────┘

                          │ Sparse vectors
                          ▼
              ┌──────────────────────────────┐
              │          SMART         │◀────────── weight ltc
              │          convert       │
              └──────────────────────────────┘

                       │ Weighted sparse vectors
                       ▼
              ┌──────────────────────────────┐
SVD ──────────────▶│          SMART         │◀────────── generate profile
              │          convert       │
              └──────────────────────────────┘

                             │
                             ▼
                          Profile
              Single dense vector in k-space
```
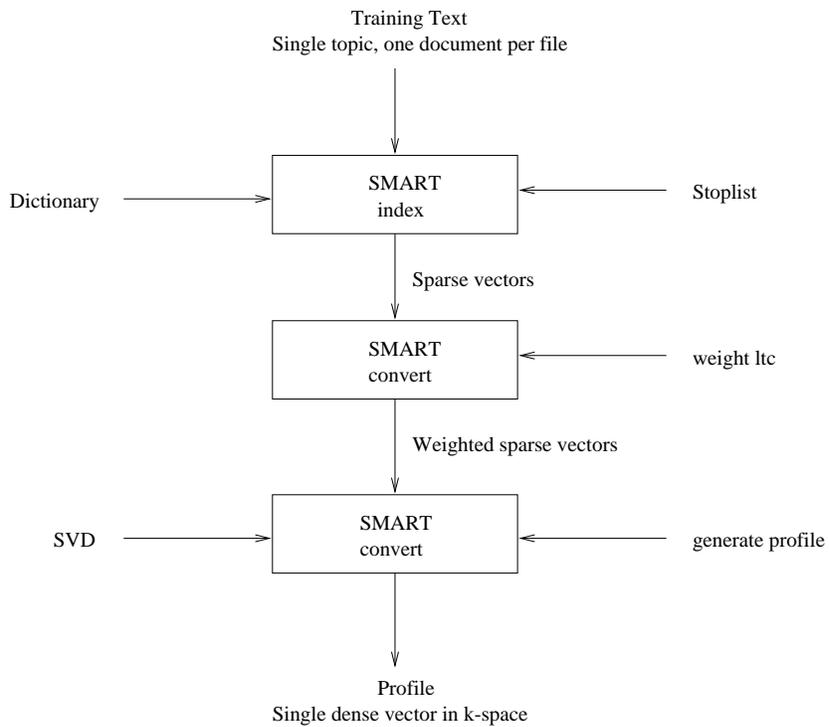
Notes: 1) Training text is Xerox English WSJ for LSI and Xerox Spanish WSJ for TT.
        2) Xerox Spanish WSJ includes untranslated English words
        3) Stoplist is English for LSI and bilingual for TT

Figure A.4: Profile training step for TT and LSC experiments.
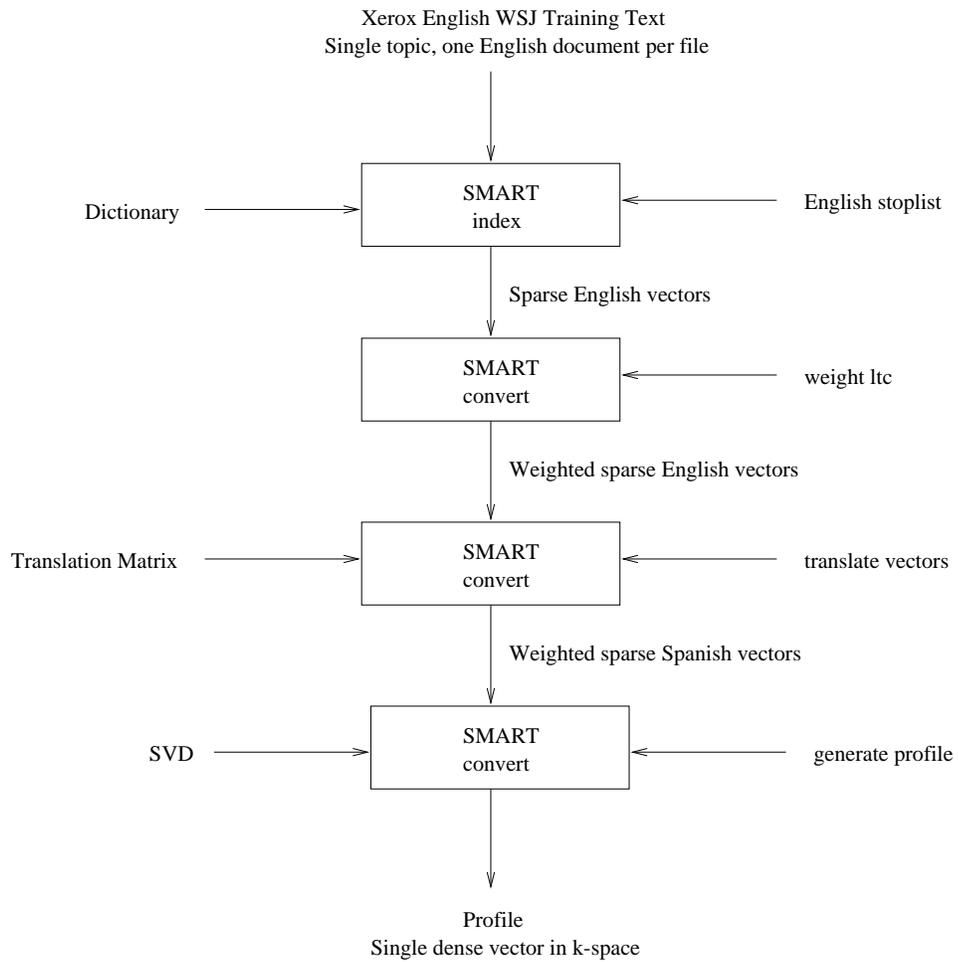
Figure A.5: Profile training step for the VT experiment.
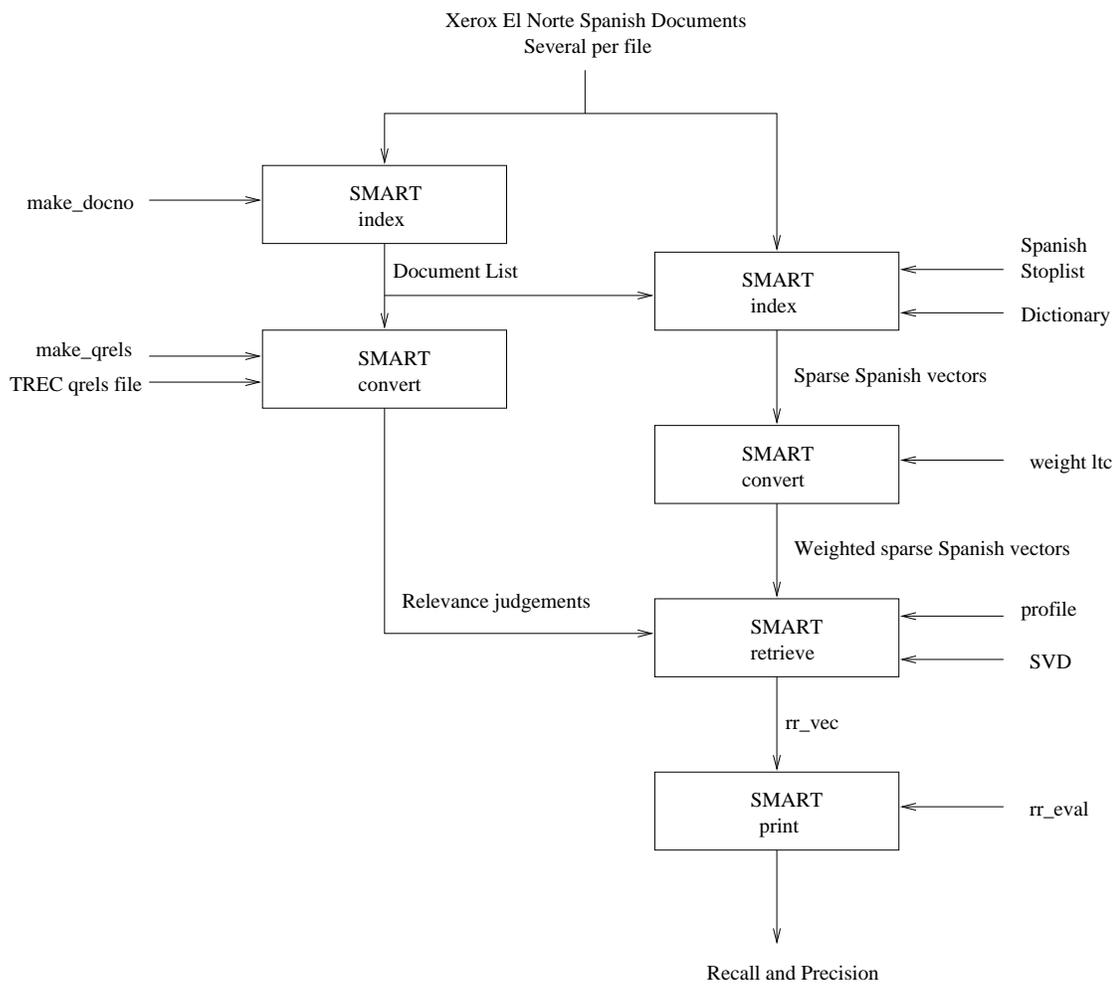
Figure A.6: Effectiveness Evaluation.

Appendix B

Use of TREC Topics

The Text REtrieval Conference (TREC) materials that we presently have available form a useful base for evaluation of cross-linguistic text filtering and retrieval systems. For cross-linguistic text retrieval we have 50 queries in both Spanish and English and a corpus 250 MB of Spanish language documents that has been scored against each of these queries. For cross-linguistic text filtering, we are able to construct an average of 200 relevant English language training documents available for each of 4 topics. For each of those topics we also have a list of the relevant documents in the 250 MB Spanish language corpus.

This appendix begins with a detailed description of the way in which the queries for the cross-linguistic retrieval evaluation, the training data for the cross-linguistic filtering evaluation, and the relevance judgments for the Spanish language corpus can be obtained from existing data. The design of the evaluation process is then presented. Finally the evaluation process will be summarized and placed in the context of a multi-system comparison.

B.1 Terminology

In TREC, ranked text retrieval is called "adhoc retrieval" and the query is referred to as a "topic specification." Ranked text filtering is referred to as "routing" and the information need representation is (unfortunately) referred to as a "query." In order to prevent any confusion about the sense of in which the word "query" is being used I will avoid further use of that word, referring instead to a "topic specification" in the case of retrieval and an "information need representation" in the case of filtering. Another unfortunate choice of terminology is the use of the term "filtering" in TREC to refer to "binary text filtering" in which a yes/no decision must be made as each new document is presented and no batch processing is allowed. Fortunately it is not necessary to discuss binary text filtering further in this paper because we are presently only considering ranked text filtering techniques. TREC uses a corpus originally developed for the DARPA TIPSTER project, so the TREC corpus is often referred to as the TIPSTER corpus.

B.1.1 English Language Training Data

One byproduct of the annual TREC is a set of relevance judgments for some portion of the TIPSTER corpus against 250 different topics. In what is known as a "pooled relevance assessment methodology," for each topic every document that highly ranked by any participating system is evaluated by professional relevance assessors for relevance to that topic's specification. These relevance assessments produce binary judgments (relevant/not relevant).

In the adhoc text retrieval system evaluation, one of two principal portions

of TREC, each system ranks a set of previously known documents on the basis of a set of 50 previously unknown queries. Thus, the adhoc text retrieval portion of each TREC conference produces, in effect, a matrix with 50 rows (one for each topic) and one column for each document in which each entry is R (relevant), N (not relevant), or U (unknown relevance). This process produces what are known as "complete" judgments because it is assumed for purposes of reporting that nearly every relevant document will have been highly ranked by some participating system. When working with a pooled relevance assessment methodology it is common to report precision (1.0 - false alarm rate) and recall (1.0 - false dismissal rate) based on the assumption that documents for which no relevance judgments are available (U) are actually not relevant (N). This assumption results in an accurate computation of precision and computation of an upper bound on recall.

In the other major portion of the TREC evaluation, the text routing evaluation, the matrix produced as a byproduct of the previous year's adhoc text retrieval evaluation is typically made available as training data for participating systems, and each system uses this information and the topic specification to build a representation of the information need. Then a set of previously unseen data is ranked by each system and every document that is highly ranked for any topic by any participating system is evaluated by the professional relevance assessors. This process extends the "complete" matrix produced in the previous year's adhoc text retrieval to a new data set.

The TIPSTER document collection is made up of three TIPSTER CDROMs (known as "TREC disks 1–3") and material which was added for TREC-4 (which is known as TREC disk 4). The documents in each set are drawn from some

subset of seven different sources. Table B.1 shows the available English language training data. Blanks in the table indicate that that material from the specified source was not included on that collection. We have not been able to easily determine when the Department of Energy material on Disk 1 was created.

Source	Years from which the material was drawn			
	Disk 1	Disk 2	Disk 3	Disk 4
AP Newswire Articles	1989	1988	1990	
Computer Select	1989-1990	1989-1990	1991-1992	
Wall Street Journal	1987-1989	1990-1992		
Federal Register	1989	1988		1994
Department of Energy	Various			
Patents			1983-1991	
San Jose Mercury News			1991	
Congressional Record				1993
Financial Times				1991-1994

Table B.1: Contents of the TREC Disks.

With the completion of TREC-4 in November of 1995, several "complete" matrices are now available. There have been a few departures from the typical TREC evaluation pattern described above, so the combined 250 row matrix actually has the form illustrated in table B.2. Blank entries indicate that combination of topics and documents have not been evaluated. "Partial" indicates that only a random subset of the documents were evaluated (this procedure was used to seed the TREC-1 text routing evaluation). "23 Complete" indicates that complete evaluations were performed for 23 of the topics in this set (50 topics scattered throughout the first 200 were used in the TREC-4 routing evaluation)

Topics	Disk 1	Disk 2	Disk 3	Disk 4
1–50	Partial	Complete		23 Complete
51–100	Complete	Complete	Complete	9 Complete
101–150	Complete	Complete	Complete	10 Complete
151–200	Complete	Complete		8 Complete
201–250		Complete	Complete	

Table B.2: Availability of TREC relevance judgements.

B.1.2   Spanish Language Evaluation Data

The TREC-3 and TREC-4 conferences also included a "special interest track" on multilingual text retrieval. The corpus for that track is approximately 250 MB of 1992 articles from the Spanish language Mexican Newspaper "El Norte." No additions to the corpus were made the second year. Although referred to as the "multilingual track," the evaluation is actually a monolingual (Spanish only) adhoc text retrieval evaluation. For TREC-3 there were 25 Spanish language topics, and another 25 were added for TREC-4. No text routing evaluation has been performed using this material, but TREC-4 multilingual track participants were asked to submit results for all 50 topics in order to expand the set of documents for which relevance judgments were available for the original 25 topics. The "El Norte" corpus and the 50 topic descriptions were distributed by FTP to participants in TREC-4.

Manually constructed English language translations of the Spanish language queries were constructed by New Mexico State University and others, and we have one complete set of translations. These translations have not been used in the TREC multilingual track evaluation, but the translations of the first 25 queries has been standardized by NIST. NIST plans to issue standard transla-

tions for the remaining 25 queries. At present, we are using manually retyped versions of the New Mexico State translations for queries SP26-SP50 since the New Mexico State translations are available in hardcopy only.

B.1.3   Topic Similarity Across Languages

Four Spanish language queries that are very similar (although not identical) to one or more English language queries which have been evaluated for relevance against the TIPSTER corpus are shown in table 5.2 in Chapter 5.

The most pervasive problem in these collections is that every one of the 50 Spanish language topics has its scope explicitly restricted to Mexico in the text of the topic description. Unfortunately, none of the English language TREC topics have a scope explicitly restricted to Mexico. But since the Mexican newspaper articles that would be judged relevant to the similar English language topics would almost certainly include some discussion of the impact of the reported information on Mexican citizens, the topics should be interchangeable across languages for all practical purposes.

As a point of interest, the eight Spanish language topics shown in table B.3 have at least one related English language topic that is either significantly more specific or partially overlaps with the Spanish language topic in some other way. It may be possible to use the available relevance data for these topic pairs in some other way, but they are not addressed further in this presentation.

B.1.4   Design of the Evaluation Process

With the material presently in hand we can train a cross-linguistic text routing system using the known relevant English language documents for topic 128. We

Spanish Language Topic		English Language Topic	
SP3	Mexico City Pollution	012	Water Pollution
SP14	PEMEX Oil Monopoly	088	Crude Oil Prices
		090	Oil and Gas Reserves
SP15	Mexico-US Fishing Dispute	077	Poaching
		182	Commercial Overfishing
SP18	Foreign Car Makers in Mexico	219	US Car Imports & Exports
SP21	Mexican Textile Employment	200	US Textile Import Impact
SP23	Foreign Investment in Mexico	006	Third World Debt Relief
		120	Economics of Terrorism
SP24	AIDS Prevention in Mexico	010	AIDS Treatments
SP39	Mexican Agriculture	016	Agro-chemical Marketing
		043	Agro-chemical Control

Table B.3: Less closely related English and Spanish TREC topics.

can then present the entire "El Norte" corpus to the system to identify those documents most closely associated with Spanish language topic SP25. Finally we can determine the precision for every value of recall between 0.1 and 1.0 in steps of 0.1 to compute a recall-precision graph. Of the available training data for topic 128, the Wall Street Journal collection on TREC disk 2 is an attractive choice because it includes articles from the same year as the articles in the "El Norte" collection. The AP Newswire articles on disks 1, 2, and 3, the San Jose Mercury News articles on Disk 3, and the Wall Street Journal articles on disk 1 could also be used as sources of training data.

We can repeat the experiment for topics 022 and SP10, although the results in that case will only be directly comparable to the results of the 128/SP25 run for documents on disk 2 because topic 022 has only been completely evaluated against the corpora on disk 2. This consideration makes the Wall Street Journal (and perhaps the AP newswire collection) on disk 2 the logical choice because

it's use will enhance comparability across queries. We can then repeat the experiment a third time for topics 008 and SP22 (which have the same limitation to disk 2) and finally for the union of topics 122, 124 and 124 with topic SP47.

A larger number of topic pairs would result in a more reliable evaluation of "average" performance, but 4 topic pairs should be adequate to gain an appreciation for the relative performance of several algorithms because performance across algorithms can be compared on a topic by topic basis.

B.2 TREC Topic Descriptions

The titles in section B.1.3 are abstracted from the complete topic specifications to illustrate the degree of similarity, and are not the actual TREC topic titles. English translations produced by New Mexico State University for each of the 4 Spanish language topics are reproduced here, along with the corresponding English language TREC topic specifications. The level of detail in TREC topic specifications has been consistently decreasing, so higher numbered topics generally have considerably shorter specifications.

```
<num>  Number:  SP10


<title>  Topic:  Mexico is an important transit country

                  in the war against narcotics


<desc> Description:


Mexico is important to the narcotraffickers of Colombia

and Peru as an entry point for the U.S.  How is Mexico

used as a transit country?


<narr>


The document should indicate methods used by narcotraffickers

to utilize Mexico as a transit country for

getting drugs into the U.S.  It should include specific
```

examples and locations and measures for stopping this
activity.

</top>

- -

<top>

<head> Tipster Topic Description

<num> Number: 022

<dom> Domain: Law and Government

<title> Topic: Counternarcotics

<desc> Description:

Document will announce countermeasures to curb the production
of illegal drugs abroad or to curb the entry of illegal drugs
into the U.S.

<narr> Narrative:

To be relevant, a document will report measures taken by the
U.S. Government either to curb production of drugs, to curb
entry into the U.S. of drugs, or to prosecute those involved
in drug trafficking, laundering of drug money, or racketeering
related to drugs.

<con> Concept(s):

1. cocaine, heroin, opium, coca, poppies, hemp, marijuana,
crack, hashish

2. smuggling, cartel, traffic, drug lords

3. producers, refineries, crops

4. Colombia, Peru, Golden Triangle, Burma, Thailand, Turkey

<fac> Factor(s):

<nat> Nationality: U.S.

</fac>

<def> Definition(s):

```
</top>



-----------------------------------------------------------------



<top>



<num> Number: SP22



<title>  Topic: Recent levels of inflation in Mexico



<desc> Description:



The inflationary process in Mexico has been followed carefully

in recent years.  To be relevant the document must include the

level of inflation for a specific period with future prospects.



<narr>  Narrative:



The document should give the national inflation rate in Mexico

for a specific period during the last few years.  It should

also include predictions of future inflation rates and ways to

control them.



</top>
```

- -

<top>

<head> Tipster Topic Description

<num> Number: 008

<dom> Domain: International Economics

<title> Topic: Economic Projections

<desc> Description:

Document will contain quantitative projections of the future
value of some economic indicator for countries other than the
U.S.

<narr> Narrative:

To be relevant, a document must include a projection of the
value of an economic indicator (e.g., gross national product
(GNP), stock market, rate of inflation, balance of payments,
currency exchange rate, stock market value, per capita income,
etc.) for the coming year, for a country other than the United

States.

<con> Concept(s):

1. inflation, stagflation, indicators, index

2. signs, projection, forecast, future

3. rise, shift, fall, growth, drop, expansion, slowdown, recovery

4. %, billions

5. Not U.S.

<fac> Factor(s):

<nat> Nationality: Not U.S.

<time> Time: Future

</fac>

<def> Definition(s):

```
</top>



----------------------------------------------------------------



<top>



<num> Number:  SP25



<title>  Topic: Program for Privatization of Mexico's Public

                 Enterprises



<desc> Description:



Mexico's privatization program is considered one of the most

successful in Latin America.  The document should describe the

process of privatization of public companies in Mexico.



<narr>



To be relevant, the document should mention the Mexican public

enterprise that has been privatized, including results and

predictions of other sectors that might be privatized.



</top>
```

- -

<top>

<head> Tipster Topic Description

<num> Number: 128

<dom> Domain: International Economics

<title> Topic: Privatization of State Assets

<desc> Description:

Document discusses a current or future sale to the private
sector, by a government or government entity, of a business,
businesses, or shares of a business owned by the state.

<smry> Summary:

Document discusses a current or future sale to the private
sector, by a government or government entity, of a business,
businesses, or shares of a business owned by the state.

<narr> Narrative:

A relevant document will discuss a completed, ongoing, or proposed sale to the private sector (e.g., offer to the public, or to a selected investor), by a government or government entity, of a business, businesses, or shares of a business owned by that government or government entity. The business and the government or government entity must be identified. A document which discusses privatization in general, privatization plans or a privatization program, but does not refer to a specific sale, is NOT relevant.

<con> Concept(s):

1. privatization effort, privatization plan, denationalization

2. private sector

3. sale, sell, offer

4. stake, shares, stock, company

<fac> Factor(s):

<time> Time: Current

<time> Time: Future

</fac>

<def> Definition(s):

</top>

<top>

<num> Number: SP47

<desc> Description:

Does Mexico have research programs for the cause of cancer?

</top>

- -

<top>

<head> Tipster Topic Description

<num> Number: 123

<dom> Domain: Medical & Biological

<title> Topic: Research into & Control of Carcinogens

<desc> Description:

Document will report on studies into linkages between
environmental factors or chemicals which might cause cancer,
and/or it will report on governmental actions to identify,
control, or limit exposure to those factors or chemicals which
have been shown to be carcinogenic.

<smry> Summary:

Document will report on studies into linkages between
environmental factors or chemicals which might cause cancer,
and/or it will report on governmental actions to identify,
control, or limit exposure to those factors or chemicals which
have been shown to be carcinogenic.

<narr> Narrative:

A relevant document will report on research into linkages
between cancer and environmental hazards and/or the efforts of
governments to limit exposure of their people to carcinogens.
The governmental action may be of any category, e.g. entry into
international agreements, enactment of domestic laws, issuance
of administrative regulations, support of carcinogen research,
air and soil sampling, launching of public education campaigns,
etc.

<con> Concept(s):

1. cancer, carcinogen

2. treaty, agreement, law, regulation, study, research,
education, Super Fund

<fac> Factor(s):

<def> Definition(s):

</top>

References

[1] Advanced Research Projects Agency, "FY 1994 SBIR solicitation, Phase I award abstracts, ARPA projects," Defense Technical Information Center, 8725 John J. Kingman Road, Suite 0944, Ft. Belvior, VA 22060, 1994, ftp://ftp.dtic.dla.mil/pub/sbir/arpa94sbir_awds.

[2] Belal Mustafa Abu Ata, Tengku Mohd. Tengku Sembok, and Mohammed Yusoff, "SISDOM: a multilingual document retrieval system," *Asian Libraries*, vol. 4, no. 3, pp. 37–46, Sept. 1995.

[3] Derek Austin, "Progress towards standard guidelines for the construction of multilingual thesauri," in *Third European Congress on Information Systems and Networks*, Commission on the European Communities, Ed. May 1977, vol. 1, pp. 341–402, Verlag Dokumentation.

[4] Paul E. Baclace, "Competitive agents for information filtering," *Communications of the ACM*, vol. 35, no. 12, pp. 50, Dec. 1992.

[5] Nicholas J. Belkin and W. Bruce Croft, "Information filtering and information retrieval: Two sides of the same coin?," *Communications of the ACM*, vol. 35, no. 12, pp. 29–38, Dec. 1992.

[6] Heiner Benking and Ulrich Kampffmeyer, "Harmonization of environmental meta-information with a thesaurus-based multi-lingual and multi-medial information system," in *AIP Conference Proceedings 283, Earth and Space Science Information Systems*, Arthur Zygielbaum, Ed. American Institute of Physics, 1992, pp. 688–695.

[7] M. Berry and P. Young, "Using latent semantic indexing for multilanguage information retrieval," *Computers and the Humanities*, vol. 29, no. 6, pp. 413–429, Dec. 1995.

[8] M. W. Berry and R. D. Fierro, "Low-rank orthogonal decompositions for information retrieval applications," *Numerical Linear Algebra with Applications*, 1996, To appear. http://www.cs.utk.edu/~library/TechReports/1995/ut-cs-95-284.ps.Z.

[9] Michael Berry, Theresa Do, Gavin O'Brien, Vijay Krishna, and Sowmini Varadhan, "SVDPACKC (version 1.0) user's guide," Tech. Rep. UT-CS-93-194, University of Tennessee, Knoxville, Apr. 1993, http://www.cs.utk.edu/~library/TechReports/1993/ut-cs-93-194.ps.Z.

[10] Micheal W. Berry, Susan T. Dumais, and Gavin W. O'Brien, "Using linear algebra for intelligent information retrieval," *SIAM Review*, vol. 37, no. 4, pp. 573–595, Dec. 1996, http://www.cs.utk.edu/~library/TechReports/1994/ut-cs-94-270.ps.Z.

[11] D. C. Blair, *Language and Representation in Information Retrieval*, Elsevier, Amsterdam, 1990.

[12] Paul Blake, "The MenUSE system for multilingual assisted access to online databases," *Online Review*, vol. 16, no. 3, pp. 139–145, June 1992.

[13] P. Bollmann and E. Konrad, "Automatic association methods in the construction of interlingual thesauri," in *EURIM II A European Conference on the Application of Research in Information Science and Libraries*, W. E. Batten, Ed. Aslib, 1976, pp. 152–155.

[14] T. F. Bowen, G. Gopal, G. Herman, T.Hickey, K.C. Lee, W. H. Mansfield, J. Raitz, and A. Weiribnrib, "The datacycle architecture," *Communications of the ACM*, vol. 35, no. 12, pp. 71–80, Dec. 1992.

[15] Robert S. Brewer and Philip M. Johnson, "Toward collaborative knowledge management within large, dynamically structured information systems," Tech. Rep. ICS-TR-92-22, University of Hawaii, Department of Information and Computer Sciences, Honolulu, Oct. 1994, ftp: //ftp.ics.hawaii.edu/pub/tr/ics-tr-94-02.ps.Z.

[16] Peter F. Brown, John Cocke, Steven A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin, "A statistical approach to machine translation," *Computational Linguistics*, vol. 16, no. 2, pp. 79–85, June 1990.

[17] Peter F. Brown, Steven A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, June 1993.

[18] Chris Buckley, Gerard Salton, James Allan, and Amit Singhal, "Automatic query expansion using SMART: TREC 3," in *Overview of the Third Text REtrieval Conference (TREC-3)*, D. K. Harman, Ed. Nov. 1994, pp. 69–80, NIST, http://potomac.ncsl.nist.gov/TREC/trec3.papers/cornell.new.ps.

[19] C. Cacares, "Russian-Spanish multisubject computer dictionary," *Automatic Documentation and Mathematical Linguistics*, vol. 20, no. 2, pp. 122–125, 1986, English translation from Russian.

[20] Roberto Cencioni and Ewan Klein, "Telematics programme 1991-1994 Language Research & Engineering (LRE) an overview," Directorate General XIII, Commission of the European Communities, June 1994.

[21] Vinod Chachra, "Subject access in an automated multithesaurus and multilingual environment," in *Automated Systems for Access to Multilingual and Multiscript Library Materials*, Sally McCallum and Monica Ertel, Eds. International Federation of Library Associations and Institutions (IFLA), Aug. 1993, pp. 63–76, K. G. Saur.

[22] Larry Tzuchu Chang, "Clustering properties of quadratic neural networks for medical applications," M.S. thesis, University of Maryland, College Park, 1991.

[23] David L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–88, Feb. 1981.

[24] Ewa Chmielewska-Gorczyca and Waclaw Struk, "Translating multilingual thesauri," in *Proceedings of the First European ISKO Conference*, Pavla

Stančiková and Ingetraut Dahlberg, Eds. International Society for Knowledge Organization, Sept. 1994, pp. 150–155, Indeks Verlag.

[25] David A. Cooper and Kenneth P. Birman, "Preserving privacy in a network of mobile computers," in *Proceedings of the 1995 IEEE Symposium on Security and Privacy*. IEEE Computer Society, May 1995, pp. 26–38, http://cs-tr.cs.cornell.edu.

[26] W. B. Croft, J. Broglio, and H. Fujii, "Applications of multilingual text retrieval," in *Proceedings of the Twenty-Ninth Annual Hawaii International Conference on System Sciences*, 1995, pp. 98–107.

[27] C. J. Crouch, "An approach to the automatic construction of global thesauri," *Information Processing and Management*, vol. 26, no. 5, pp. 629–640, Sept. 1990.

[28] Kenn T. Dahl, "OCR voting overview," in *1995 Symposium on Document Image Understanding Technology*, College Park, MD, Oct. 1995, pp. 107–112, University of Maryland Center for Automation Research.

[29] Mark Davis and Ted Dunning, "A TREC evaluation of query translation methods for multi-lingual text retrieval," in *The Fourth Text Retrieval Conference (TREC-4)*, D. K. Harman, Ed. Nov. 1995, NIST, http://crl.nmsu.edu/ANG/MWD/Book2/trec4.ps.

[30] Mark W. Davis and Ted E. Dunning, "Query translation using evolutionary programming for multi-lingual information retrieval," in *Proceedings of the Fourth Annual Conference on Evolutionary Programming*, Mar. 1995, http://crl.nmsu.edu/ANG/MWD/Book2/evolmltr1.ps.gz.

[31] Mark W. Davis and Ted E. Dunning, "Query translation using evolutionary programming for multilingual information retrieval II," in *Proceedings of the Fifth Conference on Evolutionary Programming*, Mar. 1996, http://crl.nmsu.edu/ANG/MWD/Book2/ep96.ps.

[32] Carmen Lopez de Sosoaga, "Multilingual access to documentary data base," in *Proceedings of a Conference on Intelligent Text and Image Handling (RIAO 91)*, A. Lichnerowicz, Ed., Amsterdam, Apr. 1991, pp. 774–788, Elsevier.

[33] Nicholas DeClaris, "Optimization of ellipsoidal-clusters for medical differential diagnosis," in *Proceedings of the 8th IFIP Conference on Optimization Techniques*, Josef Stoer, Ed. International Federation for Information Processing, Sept. 1977, Springer-Verlag, Abstract of an invited address.

[34] Nicholas DeClaris, Donna Harman, Christos Faloutsos, Susan Dumais, and Douglas Oard, "Information filtering and retrieval: Overview, issues and directions," in *Proceedings of the 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Norman F. Sheppard, Jr., Murray Eden, and Gideon Kantor, Eds. Nov. 1994, vol. 1, pp. 42a–49a, IEEE, http://www.ee.umd.edu/medlab/filter/papers/balt.ps.

[35] Nicholas DeClaris, Robert Newcomb, and Olusola Ijaola, "Novel uses of quadratic surfaces for medical diagnosis," in *Proceedings of the Fourteenth Annual Allerton Conference on Circuit and System Theory*, Sept. 1976.

[36] Nicholas DeClaris and Mu-Chun Su, "A self learning neuro-fuzzy system," in *Hubrid Systems II*, Panos Antsaklis, Wolf Kohn, Anil Nerode, and Shankar Sastry, Eds., pp. 106–127. Springer, Berlin, 1995.

[37] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, Sept. 1990, http://superbook.bellcore.com/~std/papers/ JASIS90.ps.

[38] Peter J. Denning, "Electronic junk," *Communications of the ACM*, vol. 25, no. 3, pp. 163–165, Mar. 1992.

[39] Barbara Denton, "Ten ways to control dialog alert costs," *Online*, vol. 19, no. 2, pp. 47–48, Mar. 1995.

[40] C. P. R. Dubois, "Free text vs. controlled vocabulary; a reassessment," *Online Review*, vol. 11, no. 4, pp. 243–253, Aug. 1987.

[41] S. T. Dumais, "Latent Semantic Indexing (LSI): TREC-3 report," in *Overview of the Third Text REtrieval Conference*, Donna Harman, Ed. Nov. 1994, pp. 219–230, NIST, http://potomac.ncsl.nist.gov/TREC/.

[42] Susan T. Dumais, "Improving the retrieval of information from external sources," *Behavior Research Methods, Instruments and Computers*, vol. 23, no. 2, pp. 229–236, May 1991.

[43] Ted E. Dunning and Mark W. Davis, "Multi-lingual information retrieval," Memoranda in Cognitive and Computer Science MCCS-93-252, New Mexico State University, Computing Research Laboratory, Feb. 1993, http://crl.nmsu.edu/ANG/MWD/Book2/mltr.ps.gz.

[44] D. A. Evans, S. K. Handerson, I. A. Monarch, J. Pereiro, L. Delon, and W. R. Hersh, "Mapping vocabularies using "latent semantics"," Tech.

Rep. CMU-LCL-91-1, Carnegie Mellon University, Laboratory for Computational Linguistics, July 1991.

[45] Christos Faloutsos and King-Ip Lin, "FastMap: A fast algorithm for indexing, data mining and visualization of traditional and multimedia datasets," in *ACM SIGMOD 95 Conference Proceedings*, 1995, http://www.cs.umd.edu/Document/UMCP-CSD:CS-TR-3383.

[46] Christian Fluhr, "Multilingual information retrieval," in *Survey of the State of the Art in Human Language Technology*, Ronald A Cole, Joseph Mariani, Hans Uszkoreit, Annie Zaenen, and Victor Zue, Eds., pp. 391–305. Center for Spoken Language Understanding, Oregon Graduate Institute, 1995, http://www.cse.ogi.edu/CSLU/HLTsurvey/ch8node7.html.

[47] Christian Fluhr and Khaled Radwan, "Fulltext databases as lexical semantic knowledge for multilingual interrogation and machine translation," in *Proceedings of the East-West Conference on Artificial Intelligence (EWAIC '93)*, Patrick Brezillon and Vadim Stefanuk, Eds., Moscow, Sept. 1993, Association for Artificial Intelligence of Russia, pp. 124–128, ICSTI.

[48] Peter W. Foltz, "Using latent semantic indexing for information filtering," in *Conference on Office Information Systems*, Frederick H. Lochovsky and Robert B. Allen, Eds. Apr. 1990, pp. 40–47, ACM, http://www-psych.nmsu.edu/~pfoltz/cois/filtering-cois.html.

[49] Peter W. Foltz and Susan T. Dumais, "Personalized information delivery: An analysis of information filtering methods," *Communications of the ACM*, vol. 35, no. 12, pp. 51–60, Dec. 1992, http://www-psych.nmsu.edu/~pfoltz/cacm/cacm.html.

[50] Office for Official Publications of the European Communities, *Thesaurus EUROVOC Volume 3: Multilingual version*, Luxembourg, 1995.

[51] William B. Frakes and Ricardo Baeza-Yates, Eds., *Information Retrieval: Data Structures and Algorithms*, Prentice Hall, Englewood Cliffs, NJ, 1992.

[52] Jerome H. Friedman, "Regularized discriminant analysis," *Journal of the American Statistical Association*, vol. 84, no. 405, pp. 165–175, Mar. 1989.

[53] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais, "The vocabulary problem in human-system communication," *Communications of the Association for Computing Machinery*, vol. 30, no. 11, pp. 964–971, Nov. 1987.

[54] George W. Furnas, Scott Deerwester, Susan T. Dumais, Thomas K. Landauer, Richard A. Harshman, Lynn A. Streeter, and Karen E. Lochbaum, "Information retrieval using a singular value decomposition model of latent semantic structure," in *11th International Conference on Research and Development in Information Retrieval*. June 1988, pp. 465–480, ACM.

[55] Osamu Furuse and Hitoshi Iida, "An example-based method for transfer-driven machine translation," in *Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, June 1992, pp. 139–150.

[56] David Goldberg, David Nicholas, Brian M. Oki, and Douglas Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, Dec. 1992.

[57] Louise Guthrie, James Pustejovsky, Yorick Wilks, and Brian M. Slator, "The role of lexicons in natural language processing," *Communications of the Association for Computing Machinery*, vol. 39, no. 1, pp. 63–72, Jan. 1996.

[58] Donna Harman, "The DARPA TIPSTER project," *ACM SIGIR Forum*, vol. 26, no. 2, pp. 26–28, Fall 1992.

[59] Donna Harman, "Overview of the first TREC conference," in *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Robert Korfhage, Edie Rasmussen, and Peter Willett, Eds. June 1993, pp. 36–47, ACM.

[60] Donna Harman, "Overview of the third Text REtrieval Conference (TREC-3)," in *Overview of the Third Text REtrieval Conference (TREC-3)*, D. K. Harman, Ed. NIST, 1994, pp. 1–19, U.S. Department of Commerce, NIST Special Publication 500-225. http://potomac.ncsl.nist.gov/TREC.

[61] Joseph D. Harwood, "Neural network implementation of a novel heuristic learning algorithm," M.S. thesis, University of Maryland, College Park, 1990.

[62] W. C. Hill, J. D. Hollan, D. Wroblewski, and T. McCandless, "Read wear and edit wear," in *Proceedings of ACM Conference on Human Factors in Computing Systems, CHI '92.* 1992, pp. 3–9, ACM Press.

[63] Will Hill, Mark Rosenstein, and Larry Stead, "Community and history-of-use navigation," in *Electronic Proceedings of the Second World*

Wide Web Conference '94. National Center For Supercomputer Applications, Software Development Group, Oct. 1994, Not available in print. http://community.bellcore.com/navigation/home-page.html.

[64] Lynette Hirschman, "Comparing MUCK-II and MUC-3: Assessing the difficulty of different tasks," in *Proceedings, Third Message Understanding Conference (MUC-3)*. DARPA, May 1991, pp. 25–30, Morgan Kaufmann.

[65] Edward M. Housman, "Survey of current systems for selective dissemination of information," Tech. Rep. SIG/SDI-1, American Society for Information Science Special Interest Group on SDI, Washington, DC, June 1969.

[66] Chung hsin Lin and Hsinchun Chen, "An automatic indexing and neural network approach to concept retrieval and classification of multilingual (Chinese-English) documents," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 26, no. 1, pp. 75–88, Feb. 1996, http://ai.bpa.arizona.edu/papers/chinese93/chinese93.html.

[67] David Hull, "Improving text retrieval for the routing problem using latent semantic indexing," in *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, W. Bruce Croft and C. J. van Rijsbergen, Eds. July 1994, pp. 282–291, Springer-Verlag.

[68] David A. Hull, *Information Retrieval Using Statistical Classification*, Ph.D. thesis, Stanford University, Nov. 1994, http://www.ee.umd.edu/medlab/filter/papers/hull.ps.

[69] David A. Hull and Gregory Grefenstette, "Experiments in multilingual information retrieval," in *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996, To appear. http://www.xerox.fr/grenoble/mltt/people/hull/papers/sigir96.ps.

[70] Olusola Olu. Ijaola, *An algorithmic Approach to Disease Classification and to Medical Diagnosis*, Ph.D. thesis, University of Maryland, College Park, 1977.

[71] A. Iljon, "Creation of thesauri for EURONET," in *Third European Congress on Information Systems and Networks*, Commission of the European Communities, Ed. May 1977, vol. 1, pp. 417–437, Verlag Dokumentation.

[72] Ariane Iljon, "Scientific and technical data bases in a multilingual society," *On-Line Review*, vol. 1, no. 2, pp. 133–136, June 1977.

[73] Paul S. Jacobs and Lisa F. Rau, "SCISOR: Extracting information from on-line news," *Communications of the ACM*, vol. 33, no. 11, pp. 88–97, Nov. 1990.

[74] Zhenglian Jiang, "Understanding information filtering and providing and information filtering system model," M.S. thesis, University of Missouri, Kansas City, Dec. 1993.

[75] S. Ya. Kalachkina, "Algorithmic determination of descriptor equivalents in different natural languages," *Automatic Documentation and Mathemat-*

ical Linguistics, vol. 21, no. 4, pp. 21–29, 1987, English translation from Russian.

[76] Jussi Karlgren, Kristina Hook, Ann Lantz, Jacob Palme, and Daniel Pargman, "The glass box user model for filtering," Tech. Rep. T94:09, Swedish Institute of Computer Science, July 1994, http://mars.dsv.su.se/ ~fk/if_Doc/JPfilter-filer/Glassbox1.1.ps.Z.

[77] Genichiro Kikui, Yoshihiko Hayashi, and Seiji Suzaki, "Cross-lingual information retrieval on the WWW," in *Multilinguality in Software Engineering: The AI Contribution*. European Coordinating Committee for Artificial Intelligence, Aug. 1996, To appear. http://isserv.tas.ntt.jp/ chisho/paper/9608KikuiMULSAIC.ps.Z.

[78] Hiroaki Kitano, "Multilingual information retrieval mechanism using VLSI," in *RIAO 88 Program: User-Oriented Content-Based Text and Image Handling*, A. Lichnerowicz, Ed., Mar. 1988, vol. 2, pp. 1044–1059.

[79] Robert R. Korfhage, "To see, or not to see—is that the query?," in *Proceedings of the Fourteenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval*, A. Bookstein, Y. Chiaramella, G. Salton, and V. V. Raghavan, Eds., Oct. 1991, pp. 134–141.

[80] Thomas K. Landauer and Michael L. Littman, "Fully automatic cross-language document retrieval using latent semantic indexing," in *Proceedings of the Sixth Annual Conference of the UW Centre for the New Oxford English Dictionary and Text Research*, pp. 31–38. UW Centre

for the New OED and Text Research, Waterloo Ontario, October 1990, http://www.cs.duke.edu/~mlittman/docs/x-lang.ps.

[81] Thomas K. Landauer and Michael L. Littman, "A statistical method for language-independent representation of the topical content of text segments," in *Proceedings of the Eleventh International Conference: Expert Systems and Their Applications*, Avignon France, May 1991, vol. 8, pp. 77–85.

[82] Pat Langley, *Elements of Machine Learning*, Morgan Kaufmann, San Francisco, 1996.

[83] Abraham I Lebowitz, Robert Portegies Zwart, and Helga Schmid, "Multilingual indexing and retrieval in bibliographic systems: The AGRIS experience," *Quarterly Bulletin of the International Association of Agricultural Librarians and Documentalists*, vol. 36, no. 3, pp. 187–192, 1991.

[84] Wendy Lehnert and Beth Sundheim, "A performance evaluation of text analysis technologies," *AI Magazine*, vol. 12, no. 3, pp. 81–94, Fall 1991.

[85] David Dolan Lewis, *Representation and Learning in Information Retrieval*, Ph.D. thesis, University of Massachusetts, Feb. 1992.

[86] C. S. Li, A. S. Pollitt, and M. P. Smith, "Multilingual MenUSE - a Japanese front-end for searching English language databases and vice versa," in *Proceedings of the 14th BCS IRSG Research Colloquium on Information Retrieval*. Apr. 1992, Springer-Verlag.

[87] Shoshana Loeb, "Architecting personalized delivery of multimedia information," *Communications of the ACM*, vol. 35, no. 12, pp. 39–48, Dec. 1992.

[88] B. R. Loginov and V. V. V'yugin, "Automated maintenance of a bilingual medical thesaurus on a microcomputer," *Automatic Documentation and Mathematical Linguistics*, vol. 23, no. 2, pp. 72–75, 1989, English translation from Russian.

[89] H. P. Luhn, "A statistical approach to mechanized encoding and searching of literary information," *IBM Journal of Research and Development*, vol. 1, no. 4, pp. 309–317, Oct. 1957.

[90] H. P. Luhn, "A business intelligence system," *IBM Journal of Research and Development*, vol. 2, no. 4, pp. 314–319, Oct. 1958.

[91] Thomas W. Malone, Kenneth R. Grant, Franklyn A. Turbak, Steven A. Brobst, and Michael D. Cohen, "Intelligent information sharing systems," *Communications of the ACM*, vol. 30, no. 5, pp. 390–402, May 1987.

[92] Gary Marchionini, *Information Seeking in Electronic Environments*, Cambridge University Press, Cambridge, 1995.

[93] Richard S. Marcus, "Intelligent assistance for document retrieval based on contextual, structural, interactive Boolean models," in *RIAO 94 Conference Proceedings, Intelligent Multimedia Information Retrieval Systems and Management*, Paris, Oct. 1994, vol. 2, pp. 27–43, Centre de Hautes Etudes Internationales d'Informatique Documentaire (C.I.D.).

[94] Matt Mettler, "TRW Japanese fast data finder," in *TIPSTER Text Program Phase I: Proceedings of a Workshop held at Fredricksburg, Virginia*. ARPA, Sept. 1993, pp. 113–116, Morgan Kaufmann.

[95] Kenrick Jefferson Mock, *Intelligent Information Filtering via Hybrid Techniques: Hill Climbing, Case-Based Reasoning, Index Patterns, and Genetic Algorithms*, Ph.D. thesis, University of California Davis, 1996, http://phobos.cs.ucdavis.edu:8001/~mock/infos/infos.html.

[96] Masahiro Morita and Yoichi Shinoda, "Information filtering based on user behavior analysis and best match text retrieval," in *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, W. Bruce Croft and C.J. van Rijsbergen, Eds. July 1994, pp. 272–281, Springer-Verlag, http://www.jaist.ac.jp/jaist/is/labs/shinoda-lab/papers/1994/sigir-94.ps.

[97] P. Nelson, "Breaching the language barrier: Experimentation with Japanese to English machine translation," in *15th International Online Information Meeting Proceedings*, David I Raitt, Ed. Dec. 1991, pp. 21–33, Learned Information.

[98] H. Neville, "Session V report of the English language discussion group," in *Second European Congress on Information Systems and Networks*. May 1975, pp. 162–164, Verlag Dokumentation.

[99] H. H. Neville, "Feasibility study of a scheme for reconciling thesauri covering a common subject," *Journal of Documentation*, vol. 26, no. 4, pp. 313–336, Dec. 1970.

[100] H. H. Neville, "Alternatives to conventional multilingual thesauri," in *Report of a Workshop on Multilingual Systems*, Verina Horsnell, Ed., 1975, pp. 10–12, British Library Research and Development Report 5265 HC.

[101] Douglas W. Oard and Nicholas DeClaris, "Cognitive models for text filtering," Tech. Rep. EE-TR-96-28, University of Maryland, College Park, 1996, To appear.

[102] Douglas W. Oard, Nicholas DeClaris, Bonnie J. Dorr, and Christos Faloutsos, "On automatic filtering of multilingual texts," in *Conference Proceedings, 1994 IEEE International Conference on Systems, Man, and Cybernetics*, Oct. 1994, vol. 2, pp. 1645–1650, http://www.ee.umd.edu/medlab/filter/papers/smc.ps.

[103] Douglas W. Oard, Nicholas DeClaris, Bonnie J. Dorr, Christos Faloutsos, and Gary Marchionini, "Experimental investigation of high performance cognitive and interactive text filtering," in *Conference Proceedings, 1995 IEEE International Conference on Systems, Man and Cybernetics*, Oct. 1995, http://www.ee.umd.edu/medlab/filter/papers/smc95.ps.

[104] Douglas W. Oard and Bonnie J. Dorr, "A survey of multilingual text retrieval," Tech. Rep. UMIACS-TR-96-19, University of Maryland, Institute for Advanced Computer Studies, Apr. 1996, http://www.ee.umd.edu/medlab/filter/papers/mlir.ps.

[105] Douglas W. Oard and Gary Marchionini, "A conceptual framework for text filtering," Tech. Rep. CS-TR-3643, University of Maryland, May 1996, http://www.ee.umd.edu/medlab/filter/papers/filter.ps.

[106] Gavin W. O'Brien, "Information management tools for updating an SVD-encoded indexing scheme," M.S. thesis, University of Tennessee, Knoxville, Dec. 1994, http://www.cs.utk.edu/~library/TechReports/1994/ut-cs-94-258.ps.Z.

[107] Irma Pasanen-Tuomainen, "Analysis of subject searching in the TENTTU books database," in *Proceedings of the 14th Biennial Conference of IATUL*, Jay K. Lucker, Ed. International Association of Technological University Libraries, June 1991, vol. 1, pp. 72–77.

[108] N. A. Pashchenko, S. Ya. Kalachkina, N. M. Matsak, and V. A. Pigur, "Basic principles for creating multilanguage information retrieval thesauri (experience with implementing GOST 7.24-80)," *Automatic Documentation and Mathematical Linguistics*, vol. 16, no. 3, pp. 30–36, 1982, English translation from Russian.

[109] D. Pelissier and O. Artur, "The multilingual evolution of PASCAL," in *10th International Online Information Meeting*. Dec. 1986, pp. 113–121, Learned Information.

[110] B. R. Pevzner, "Automatic translation of English text to the language of the Pusto-Nepusto-2 system," *Automatic Documentation and Mathematical Linguistics*, vol. 3, no. 4, pp. 40–48, 1969, English translation from Russian.

[111] B. R. Pevzner, "Comparative evaluation of the operation of the Russian and English variants of the "Pusto-Nepusto-2" system," *Automatic Documentation and Mathematical Linguistics*, vol. 6, no. 2, pp. 71–74, 1972, English translation from Russian.

[112] V. A. Pigur, "Multilanguage information-retrieval systems: Integration levels and language support," *Automatic Documentation and Mathematical Linguistics*, vol. 13, no. 1, pp. 36–46, 1979, English translation from Russian.

[113] A. Steven Pollitt and Geoff Ellis, "Multilingual access to document databases," in *21st Annual Conference Canadian Society for Information Science*, July 1993, pp. 128–140.

[114] A. Steven Pollitt, Geoffrey P. Ellis, Martin P. Smith, Mark R. Gregory, Chun Sheng Li, and Henrik Zangenberg, "A common query interface for multilingual document retrieval from databases of the European Community institutions," in *Proceedings of the 17th International Online Information Meeting*, Dec. 1993, pp. 47–61.

[115] Stephen Pollock, "A rule-based message filtering system," *ACM Transactions on Office Information Systems*, vol. 6, no. 3, pp. 232–254, July 1988.

[116] Barbara Rad-El, "Approaches to multilanguage and multiscript issues in the ALEPH system," in *Automated Systems for Access to Multilingual and Multiscript Library Materials*, Sally McCallum and Monica Ertel, Eds. International Federation of Library Associations and Institutions, Aug. 1993, pp. 145–150, K. G. Saur.

[117] Khaled Radwan, *Vers l'Accés Multilingue en Langage Naturel aux Bases de Données Textuelles*, Ph.D. thesis, Université de Paris-Sud, Centre d'Orsay, 1994.

[118] Khaled Radwan and Christian Fluhr, "Textual database lexicon used as a filter to resolve semantic ambiguity application on multilingual information retrieval," in *Fourth Annual Symposium on Document Analysis and Information Retrieval*, Apr. 1995, pp. 121–136.

[119] Khaled Radwan, Frederic Foussier, and Christian Fluhr, "Multilingual access to textual databases," in *Proceedings of a Conference on Intelligent Text and Image Handling (RIAO 91)*, A. Lichnerowicz, Ed. Apr. 1991, pp. 475–489, Elsevier.

[120] Ashwin Ram, "Natural language understanding for information filtering systems," *Communications of the ACM*, vol. 35, no. 12, pp. 80–81, Dec. 1992, ftp://ftp.cc.gatech.edu/ai/ram/er-92-08.ps.Z.

[121] C. Radhakrishna Rao, *Contributions to Statistics*, Pergamon Press, Oxford, 1963.

[122] A. M. Rassinoux, R. H. Baud, and J. R. Scherrer, "A multilingual analyser of medical texts," in *Second International Conference on Conceptual Structures (ICCS 94)*, W. M. Tepfenhart, J. P. Dick, and J. F. Sowa, Eds. 1994, pp. 84–96, Springer-Verlag, http://www.hbroussais.fr/helios/doc/nlp/Rassinoux94b.html.

[123] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proceedings of the Conference on Computer Supported Cooperative Work*, Richard K. Faruta and Christine M. Neuwirth, Eds. ACM, Oct. 1994, pp. 175–186, http://www.cs.umn.edu/Research/GroupLens/cscwpaper/paper.html.

[124] E. A. Rich, "User modeling via stereotypes," *Cognitive Science*, vol. 3, pp. 329–354, July 1979.

[125] Paule Rolland-Thomas and Gérard Mercure, "Subject access in a bilingual online catalog," *Cataloging and Classification Quarterly*, vol. 10, no. 1/2, pp. 141–163, 1989.

[126] Loll Rolling, "Multilingual systems: survey of the European scene," in *Report of a Workshop on Multilingual Systems*, Verina Horsnell, Ed., Oct. 1975, pp. 4–5, British Library Research and Development Report 5265 HC.

[127] G. Salton, "Experiments in multi-lingual information retrieval," *Information Processing Letters*, vol. 2, no. 1, pp. 6–11, Mar. 1973, TR 72-154 at http://cs-tr.cs.cornell.edu.

[128] Gerard Salton, "Automatic processing of foreign language documents," *Journal of the American Society for Information Science*, vol. 21, no. 3, pp. 187–194, May 1970.

[129] Gerard Salton and Michael J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill, New York, 1983.

[130] Mark Sanderson, "Word sense disambiguation and information retrieval," in *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, W. Bruce Croft and C. J. van Rijsbergen, Eds. July 1994, pp. 142–151, Springer-Verlag, http://www.dcs.gla.ac.uk/ir/papers/Postscript/sanderson94b.ps.gz.

[131] Tefko Saracevic, "Measuring the degree of agreement between searchers," in *Proceedings of the 47th ASIS Annual Meeting*, Barbara Flood, Joanne Witiak, and Thomas H. Hogan, Eds. American Society for Information Science, Oct. 1984, vol. 21, pp. 227–230, Knowledge Industry Publications.

[132] Hinrich Schütze, David A. Hull, and Jan O. Pedersen, "A comparison of classifiers and document representations for the routing problem," in *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Edward A. Fox, Peter Ingwersen, and Raya Fidel, Eds., July 1995, pp. 229–237.

[133] F. Semturs, "Information retrieval from documents in multilingual textual data banks," in *Third European Congress on Information Systems and Networks*, Munich, May 1977, pp. 463–467, Verlag Dokumentation.

[134] Fritz Semturs, "STAIRS/TLS - a system for "free text" and "descriptor" searching," in *Proceedings of the ASIS Annual Meeting*, Everett H. Brenner, Ed. American Society for Information Science, Nov. 1978, vol. 15, pp. 295–298.

[135] Wade Shen and Joseph Garman, "Alignment of bilingual terms in parallel corpora," Tech. Rep. CS-TR-3666, University of Maryland, College Park, 1996.

[136] Páraic Sheridan and Jean Paul Ballerini, "Experiments in multilingual information retrieval using the SPIDER system," in *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Aug. 1996, To appear. http://www-ir.inf.ethz.ch/Public-Web/sheridan/papers/SIGIR96.ps.

[137] Beerud Sheth, "A learning approach to personalized information filtering," M.S. thesis, MIT, Media Lab, Feb. 1994, http://agents.www.media.mit.edu/groups/agents/papers/newt-thesis/main.html.

[138] Amit Singhal, Gerard Salton, Mandar Mitra, and Chris Buckley, "Document length normalization," Tech. Rep. TR95-1529, Cornell University, July 1995, http://cs-tr.cs.cornell.edu.

[139] M. P. Smith, A. S. Pollitt, and C. S. Li, "An evaluation of concept translation through menu navigation in the MenUSE intermediary system," in *14th Information Retrieval Colloquium*, Tony McEnery and Chris Paice, Eds. British Computer Society, Apr. 1992, pp. 38–54, Springer-Verlag.

[140] Irene Stadnyk and Robers Kass, "Modeling users' interests in information filters," *Communications of the ACM*, vol. 35, no. 12, pp. 49–50, Dec. 1992.

[141] Erwin Stegentritt, *German Analysis: Morpho-Syntax Within the Framework of the Free-Text Retrieval Project E.M.I.R.*, vol. 15, AQ-Verlag, Saarbrücken, Germany, 1994.

[142] Curt Stevens, "Automating the creation of information filters," *Communications of the ACM*, vol. 35, no. 12, pp. 48, Dec. 1992, http://www.holodeck.com/curt/mypapers/CACM-12-92.ps.

[143] Curt Stevens, *Knowledge-Based Assistance for Accessing Large, Poorly Structured Information Spaces*, Ph.D. thesis, University of Colorado, Department of Computer Science, Boulder, 1992, http://www.holodeck.com/curt/mypapers/Thesis.ps.gz.

[144] Roger E. Story, "An explanation of the effectiveness of latent semantic indexing by means of a Bayesian regression model," *Information Processing and Management*, vol. 32, no. 3, pp. 329–344, May 1996.

[145] Gilbert Strang, *Linear Algebra and its Applications*, Academic Press, New York, 1980.

[146] Mu-Chun Su, *A Novel Neural Network Approach to Knowledge Acquisition*, Ph.D. thesis, University of Maryland, College Park, 1993.

[147] Catherine Synellis, "TRANSLIB user survey report," TRANSLIB technical report, University of Patras Central Library, Rio 261 00 Patras, Greece, May 1995, http://grial.uc3m.es/~aedo/translib/UserAn.htm.

[148] M. Tallving and P. Nelson, "Japanese databases and machine translation: A question of international accessibility to Japanese databases," in *14th International Online Information Meeting Proceedings*, David I Raitt, Ed. Oxford, Dec. 1990, pp. 423–437, Learned Information.

[149] Robert S. Taylor, "The process of asking questions," *American Documentation*, vol. 13, no. 4, pp. 391–396, Oct. 1962.

[150] Douglas B. Terry, "A tour through Tapestry," in *Proceedings of the ACM Conference on Organizational Computing Systems (COOCS)*, Nov. 1993, pp. 21–30.

[151] Tung-Duong Tran-Luu, *Mathematical Concepts and Novel Heuristic Methods for Data Clustering and Visualization*, Ph.D. thesis, University of Maryland, College Park, 1996, http://www.glue.umd.edu/~duong/ sum.ps.

[152] Howard Turtle and W. Bruce Croft, "Inference networks for document retrieval," in *Proceedings of the 13th International Conference on Research and Development in Information Retrieval*, Jean-Luc Vidick, Ed. ACM SIGIR, Sept. 1990, pp. 1–24.

[153] Howard R. Turtle and W. Bruce Croft, "A comparison of text retrieval models," *The Computer Journal*, vol. 35, no. 3, pp. 279–290, June 1992.

[154] United Nations Educational, Scientific and Cultural Organization (UNESCO), "Guidelines for establishment and development of multilingual scientific and technical thesauri for information retrieval," Place de Fontenoy, Paris 7e, Dec. 1971, SC/WS/501.

[155] Pim van der Eijk, "Automating the acquisition of bilingual terminology," in *Sixth Conference of the European Chapter of the Association for Computational Linguistics*, Apr. 1993, pp. 113–119.

[156] K. I. Volodin, L. L. Gul'nitskii, R. N. Maksakova, V. F. Parkhomenko, I. F. Pozhariskii, L. V. Fedotova, and N.I. Yakovleva, "Bilingual indexing of geological documents," *Automatic Documentation and Mathematical Linguistics*, vol. 25, no. 6, pp. 43–45, 1991, English translation from Russian.

[157] Ellen M. Voorhees, Narendra K. Gupta, and Ben Johnson-Laird, "Learning collection fusion strategies," in *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Edward A. Fox, Peter Ingwersen, and Raya Fidel, Eds. July 1995, pp. 172–179, ACM Press.

[158] M.F. Wyle and H.P. Frei, "Retrieving highly dynamic, widely distributed information," in *Proceedings of the Twelfth Annual International ACMSI-GIR Conference on Research and Development in Information Retrieval*, N. J. Belkin and C.J. van Rijsbergen, Eds. June 1989, pp. 108–115, ACM.

[159] Mitchell F. Wyle, *Effective Dissemination of WAN Information*, Ph.D. thesis, LaSalle University, Mandeville, LA, 1995, http://vhdl.org/~wyle/diss/diss.html.

[160] Tak W. Yan and Hector Garcia-Molina, "Distributed selective dissemination of information," in *Proceedings of the Third International Conference on Parallel and Distributed Information Systems*. IEEE Computer Society, Sept. 1994, pp. 89–98, ftp://db.stanford.edu/pub/yan/1994/dsdi.ps.

[161] Tak W. Yan and Hector Garcia-Molina, "SIFT — A tool for wide-area information dissemination," in *Proceedings of the 1995 USENIX Technical Conference*, 1995, pp. 177–186, ftp://db.stanford.edu/pub/yan/1994/sift.ps.

[162] Paul G. Young, "Cross-language information retrieval using latent semantic indexing," Tech. Rep. CS-94-259, University of Tennessee, Knoxville, Dec. 1994, http://www.cs.utk.edu/~library/TechReports/1994/ut-cs-94-259.ps.Z.