ABSTRACT

Title of dissertation:	Matching Meaning for Cross-Language Information Retrieval
	Jianqiang Wang, Doctor of Philosophy, 2005
Dissertation directed by:	Professor Douglas W. Oard College of Information Studies

Cross-language information retrieval concerns the problem of finding information in one language in response to search requests expressed in another language. The explosive growth of the World Wide Web, with access to information in many languages, has provided a substantial impetus for research on this important problem. In recent years, significant advances in cross-language retrieval effectiveness have resulted from the application of statistical techniques to estimate accurate translation probabilities for individual terms from automated analysis of human-prepared translations. With few exceptions, however, those results have been obtained by applying evidence about the meaning of terms to translation in one direction at a time (e.g., by translating the queries into the document language).

This dissertation introduces a more general framework for the use of translation probability in cross-language information retrieval based on the notion that information retrieval is dependent fundamentally upon matching what the searcher means with what the document author meant. The perspective yields a simple computational formulation that provides a natural way of combining what have been known traditionally as query and document translation. When combined with the use of synonym sets as a computational model of meaning, cross-language search results are obtained using English queries that approximate a strong monolingual baseline for both French and Chinese documents. Two well-known techniques (structured queries and probabilistic structured queries) are also shown to be a special case of this model under restrictive assumptions.

MATCHING MEANING FOR CROSS-LANGUAGE INFORMATION RETRIEVAL

JIANQIANG WANG

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2005

Advisory Committee:

Professor Douglas W. Oard, Chair
Professor Vedat Diker
Professor James Mayfield
Professor Philip Resnik
Professor Dagobert Soergel

© Copyright by

Jianqiang Wang

2005

DEDICATION

to my parents

ACKNOWLEDGMENTS

My special thanks and sincere gratitude are due to my advisor, Dr. Douglas W. Oard, who has been inspiring and encouraging throughout my entire journey of pursuing the Ph.D. Without his dedication of countless hours of mentoring, it would have been impossible for me to succeed. Dr. Oard has not only guided me through each stage of my Ph.D. study at Maryland, but also has had great influence on the way I do research and communicate with others.

Many other people also contributed to the success of my dissertation. I feel grateful to all the members of my dissertation committee: Dr. Vedat Diker, Dr. James Mayfield, Dr. Philip Resnik, and Dr. Dagobert Soergel, who have provided valuable advice and comments. Colleagues in the Computational Linguistics and Information Processing Lab at the University of Maryland Institute for Advanced Computer Studies (UMIACS) have always been available for lively discussions of issues encountered throughout my studies. Recognition is deserved for all of them, in particular, Dr. Kareem Darwish for making the Perl Search Engine available, Dr. Okan Kolak and Adam Lopez for help in understanding statistical machine translation and in setting up GIZA++, Dr. David Chiang and Michael Subotin for providing Chinese processing tools, Dr. Gina-Anne Levow for her help for improving my programming skills in the early years of my graduate study, Dr. Daqing He for his valuable suggestions on brainstorming some of the ideas introduced in the thesis,

Mark Damrongsri for his expertise in setting up hardware and software, and for Dr. Bonnie Dorr, Dr. Amy Weinberg, and all others who have given valuable comments on an early presentation of the study. My gratitude also extends to Dr. Jinxi Xu at BBN for generously providing a statistical translation table that was used in a pilot study for my dissertation proposal, and to Dr. Fernando López Ostenero at UNED for extracting the EuroWordNet synsets used in the study.

Several research projects at UMIACS have provided graduate research assistantship for my studies. They have not only been helpful financially but also created a research environment for me to develop new ideas, learn technical expertise, and communicate and cooperate with others across disciplinary and department boundaries. Undoubtedly, my participation in the Translingual Information Detection, Extraction, and Summarization (TIDES) project in its full cycle has played a key role in choosing my dissertation topic. The Multilingual Access to Large Spoken Archives (MALACH) project prompted me to think about extending to my dissertation research to other closely related areas.

I would like to thank Dr. Eugene Garfield and the organizing committee of Beta Phi Mu for the Eugene Garfield Doctoral Dissertation Fellowship that I received in 2004. This has been a great honor, and it has certainly facilitated the experiment set up, data processing, and analysis that involved in this study.

My serious thought of pursuing a Ph.D. in information studies originated during the 1995-1996 academic year, when I was an exchange graduate student at the College of Information Studies. Thanks to Dr. Ann Prentice, former dean of the College of Information Studies, and her colleagues for the great opportunity, I gained a better understanding the field and prepared for my later graduate study and life in the United States.

Finally, I owe much to my family. My parents, brothers, and sisters have always cared about me and have been supportive for my life and study in a distant country. My wife has accompanied me throughout many late or even sleepless nights as I worked on my thesis. Her great confidence with me has become an important impetus for me to persevere through the most difficult stage of my Ph.D. study. Her unconditional dedication to our family and our baby has made it possible for me to concentrate on my study and research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Advances in computer and communication technology continue to drive the explosive growth of the Internet and the electronic information available in a variety of languages. Over the past five years, for example, Internet usage has increased 126% in the world - the usage has tripled in the Middle East, Latin America, and Africa, and continued to rapidly grow in Asia and Europe (see Figure 1.1). According to the information provided on its Web site, the world's leading Internet search engine Google indexes more than 8 billion Web pages. Yet, this represents just part of the whole World Wide Web. Although English continues to be the most widely-used language on the Internet, the use of non-English languages increases rapidly, accounting for 63% of the total Internet language usage, as the statistics showed in early February 2005. Among non-English languages, Asian languages including Chinese, Japanese, and Korean, and European languages including Spanish, German, French, and Italian are more commonly used than other non-English languages (see Figure 1.2). The Web contains information useful in solving many different types of information problems. It also presents the challenge of developing automatic com-

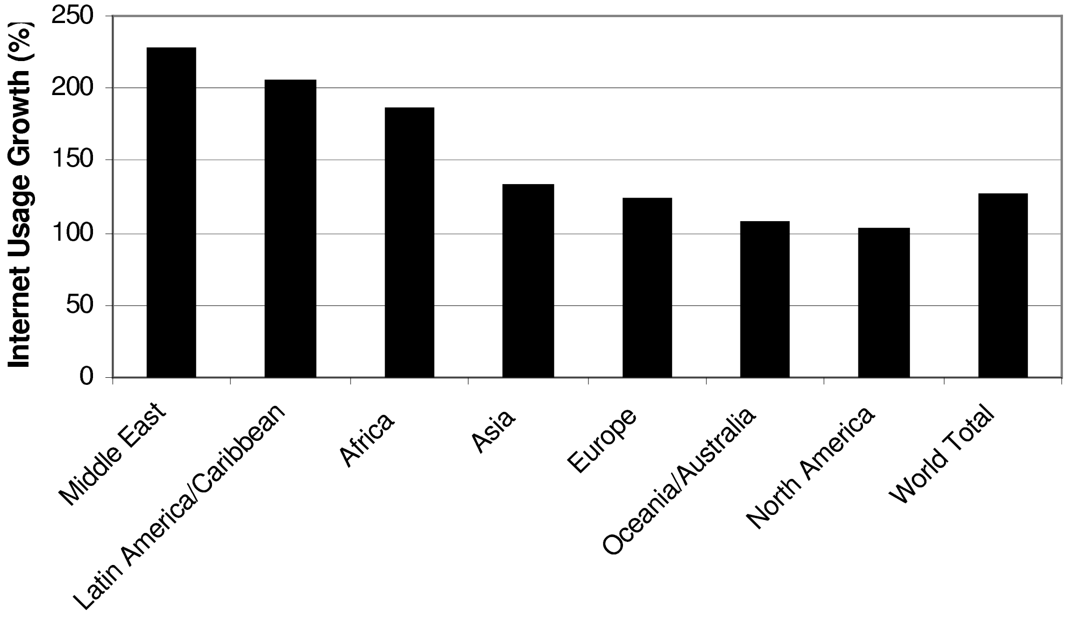


Figure 1.1: Internet usage growth (%) by World Region: 2000-2005. Internet usage information comes from data published by Nielsen//NetRatings, by International Telecommunications Union, by NICs and other reliable sources. Data was updated on February 3, 2005. Courtesy InternetWorldStats.com

puter systems to manage and provide access to information in languages unfamiliar to them. This dissertation introduces a general framework for searching text written in one language in response to information requests expressed in another language, a specific problem of Information Retrieval (IR) known as Cross-Language Information Retrieval (CLIR).

In many situations, people need to find information in an unfamiliar language or to find information in several familiar languages simultaneously. The following examples illustrate the prevalence and importance of CLIR for information users:

- A graduate student writing a survey article on the effect of nuclear power usage

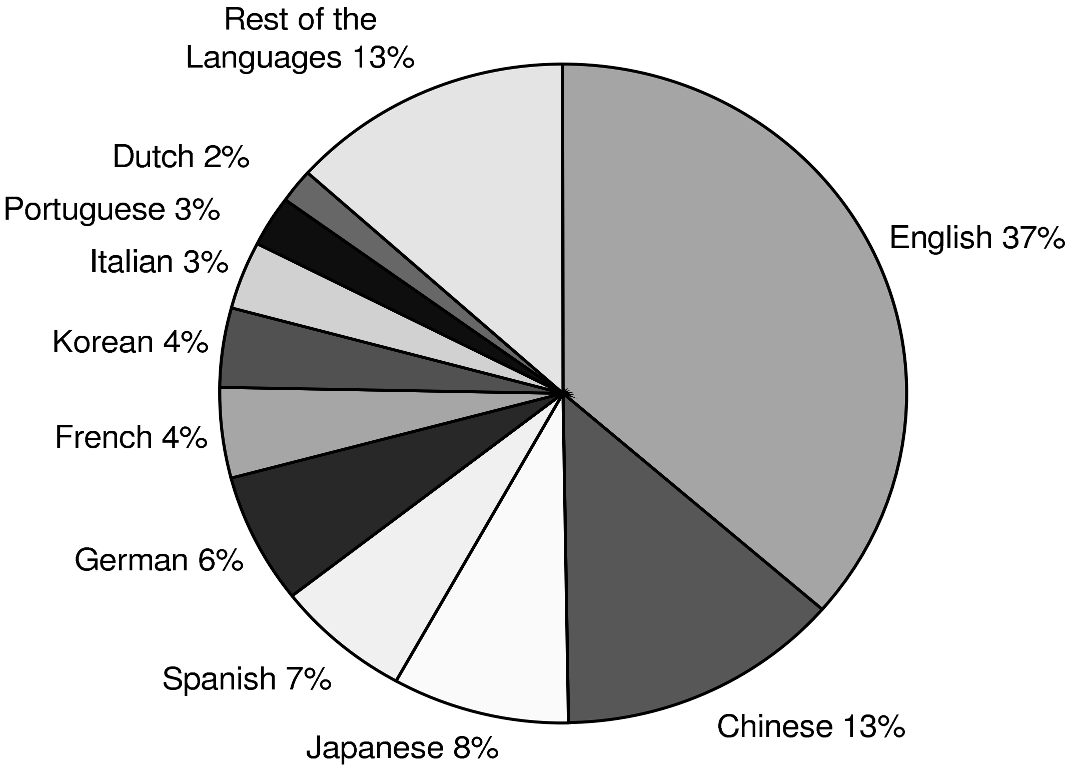


Figure 1.2: Top 10 Languages on the Internet: 2005. Data updated on February 14, 2005. Internet usage information comes from data published by Nielsen//NetRatings, International Telecommunications Union, and other reliable sources. Courtesy InternetWorldStats.com

on national economic growth in developing countries needs to find specific information about the operation, capacity, and safety of nuclear power plants in China. The majority of such information exists in newspapers published in Chinese, most of which are electronically accessible through the World Wide Web.

- The Department of Homeland Security needs information regarding institutions around the world that train non-commercial pilots. It is believed that most foreign pilot schools advertise only in their native languages.

- A historian studying the Holocaust needs to search information from a collection of interviews of Holocaust survivors, rescuers, and witnesses. The collection contains more than 50,000 audio/video interviews in 32 languages.

- A friend recounted how his advisor in China developed a novel idea for measuring distance using laser beams. The professor wants to apply for the a U.S. patent, but he does not know whether this idea has been patented in this country. He asks my friend for a favor to discover the answer.

- A high school classmate works as an anesthetist in a hospital in China. His department is choosing between two newly imported anesthesia, one from a Germany company and the other from a U.S. company. He needs publications that report the use of these two medicines. Unfortunately, he knows neither German nor English, although he speaks fluent Japanese.

- The customer service department of an international computer software marketing company frequently receive emails written in different languages, most of which requesting troubleshooting information regarding a localized version of the software.

- A military observer fluently speaking Chinese, English, and Japanese wants to find news articles published in the past decade about issues related to the Taiwan Strait.

These tasks involve searching a large collection of information items (newspaper articles, journal papers, etc.) in the same subjects as the user's interests, and the information may exist in languages unfamiliar to the users.

In this dissertation, we focus on a critical components of systems designed to handle such tasks. The component concerns the automatic matching of representations of information requests (usually called "queries") with textual documents. Since queries and documents are written in different languages, directly matching them usually fails. Therefore, some sort of "translation" should be applied. Automatic translation (that is, translation performed automatically by computers) often contains errors, which can degrade CLIR effectiveness significantly. Therefore, designing CLIR systems that are robust to erroneous translations has attracted intensive research interest.

There are generally two types of automatic translation techniques in CLIR. Dictionary-based translation attempts to find translation equivalents from machine-readable bilingual dictionaries, while corpus-based techniques extract translation knowledge from bilingual text. Each type of technique has its advantages and disadvantages. In recent years, corpus-based techniques have attracted more attention as translation probability learned from bilingual corpora has proved useful for improving cross-language search effectiveness. However, most studies on this topic have been limited to the use of unidirectional translation knowledge (for example, translation from query language to document language), and we haven't aware of any of them that used synonymy knowledge.

This dissertation introduces a general framework for cross-language information retrieval based on the notion that the goal should be to match the intended meanings of searcher and author. This perspective yields a simple computational formulation that provides a natural way of combining evidence from query and document translation, together with synonymy knowledge.

We will describe the motivation of the study in Section 1.1. Our research questions are listed in Section 1.2. Section 1.3 states the contributions of the study. The chapter ends with the layout of the dissertation in Section 1.4.

1.1 Motivation

Figure 1.3 shows a typical process of information seeking, which involves three interrelated sub-processes [59]. The entire process starts with a situation in which the user needs information to solve problems he/she encounters. In the *predict* process, users try to predict which information resources may contain the needed information, which systems they should use to find such information, and what kind of queries (including query terms and query structures) they should use. Then, the

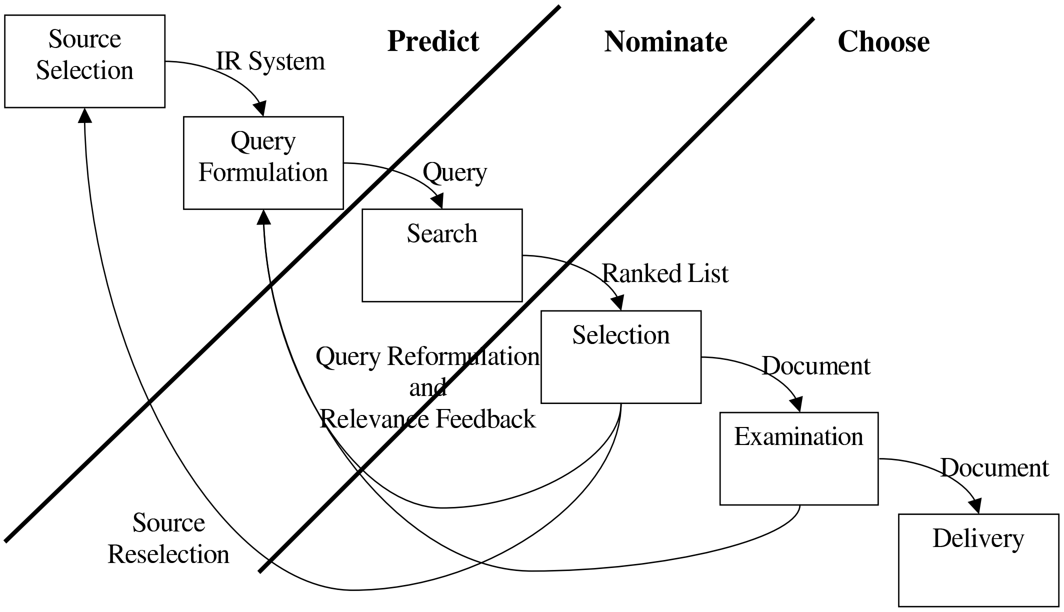


Figure 1.3: A typical information seeking process

formulated queries are fed to the *nominate* process, in which the system matches the queries automatically with documents contained in the system. Retrieved documents are displayed to the users for selection and examination in the *choice* process. Selected documents are delivered to the users so they can solve their problems. The three processes are not isolated from each other, since often what the users learned from one process could help refine actions in other processes. In this study, however, we focus on the automatic search of relevant documents, as it is a key component of any information retrieval system.

Automatic search of relevant documents occurs by matching queries and documents, and different retrieval models have been developed for it. In this study, we focus on *ranked retrieval*, that is, a relevance score is estimated for each document. Documents are ranked then by their relevance scores in decreasing order, producing a ranked list of documents for each query. In CLIR, this means for each query written in one language, we want to produce a ranked list of documents written in another language.

Traditional approaches to CLIR have either translated queries or translating documents so that queries and documents are expressed using terms in the same language; direct term matching techniques can then be employed. Both directions have weaknesses: the limited context available in (typically) short queries adds uncertainty to query translation, and computational costs can limit the extent to which context can be exploited when translating enormous document collections.

Query translation achieves the information retrieval system's goal by approximating what would have happened if the searcher had expressed their query in the document language. Document translation takes the opposite tack, approximating what would have happened if the authors had written in the query language. McCarley found that merging ranked lists generated using query translation and document translation yielded a statistically significant improvement in mean average precision compared to that achieved by either approach alone [51]. That result, which has been confirmed by others (e.g. [36]), suggests that neither query translation nor document translation alone can model adequately the probability that searchers and authors had the same meaning in mind. Boughanem et al. used bidirectional ("round trip") translation to filter potentially problematic translations, and achieved some improvement of CLIR effectiveness [5]. Inspired by these insights, we seek to explore the usefulness of applying translation knowledge in both directions in a more principled way.

Another motivation of this study involves the structured query method suggested originally by Pirkola [68]. The technique has been shown by several other studies to be effective in handling multiple translations acquired from bilingual dictionaries [53, 63, 27]. The structured query technique treats multiple translations of a query term as synonyms so each of the translations appearing in a document represents an instance of the query term in that document. Consequently, in a ranked retrieval system based on term frequency (TF) and document frequency (DF), the TF of a query term in a document can be computed by summing the TF of each of its translations in the same document, whereas the DF of the term will be the number of documents contain at least one of the target translations of the query term. Kwok later proposed to compute DF in a way similar to TF, that is, the DF of a query term is simplified as the summation of the DF of each of its translations, and no noticeable adverse effect on retrieval effectiveness has been found in a later study by Darwish and Oard [15]. In that study, Kwok's formulation also extended to handle the case in which translation probabilities are available, that is, the translation probability was used in the computation of TF and DF of each query term. They called this technique probabilistic structured queries (PSQ). They found the PSQ method could outperform the structured query method significantly, showing translation probability helps improving CLIR effectiveness.

Structured queries and PSQ are designed for translation from query language to document language. In many other situations the available translation knowledge operates in the other direction, from document language to query language. Therefore, we want to know whether similar techniques can be developed to use the document translation probability in the computation of term weight, so CLIR effectiveness can be improved in comparison to document translation without using translation probability.

The third factor motivating this study is the recent process of language modeling approaches to IR in general and to CLIR in specific [69, 29, 54, 4, 85, 43, 42]. At the core, these approaches model the probability of a query being generated by a statistical language model that is defined by a document, instead of the probability of relevance. It has been widely recognized that IR systems based on language modeling techniques can perform comparably to the best existing traditional IR systems. Furthermore, they can extend easily to CLIR by adopting a statistical document translation model in the ranking formula. In his Ph.D. dissertation study, Kraaij compared Pirkola's structured query method with several models of language modeling based CLIR and found the latter consistently outperformed the former, achieving nearly 90% of monolingual retrieval effectiveness. On the other hand, to our best knowledge, there have been no published studies in CLIR based on structured queries (including PSQ) that achieved strong monolingual effectiveness. Therefore, we are intrigued in knowing whether structured query techniques can be further improved.

The key idea supporting the meaning matching model we describe in detail in Chapter 3 is a probabilistic mapping of terms in two languages into a common meaning space. Sets of synonymous terms (synsets) are used as a computational model of meaning. This way, the probability of a term having a specific meaning can be computed by grouping its translations into synsets, based on synonymy knowledge obtained from resources such as WordNet or by reusing the statistical translation models. If knowledge of synset alignment across languages is available, full meaning matching can be occur. Otherwise, we make some simplifications so partial meaning matching becomes possible.

1.2 Research Questions

The study's high level goal is to develop a model of meaning matching for CLIR, so cross-language search effectiveness can be improved significantly. Specifically, we are interested in the following questions:

1. When using bidirectional translation knowledge and synonymy knowledge, can CLIR based on meaning matching significantly outperform the probabilistic structured query method, which uses only query translation knowledge?

2. When using translation knowledge from document language to query language alone, can CLIR based on meaning matching achieve retrieval effectiveness comparable to the probabilistic structured query method?

3. How can we establish a fair monolingual baseline to which CLIR effectiveness can be compared?

4. When using bidirectional translation knowledge and synonymy knowledge, can CLIR based on meaning matching achieve effectiveness comparable to monolingual effectiveness?

5. How does the effectiveness of the meaning matching model change according to the number of translation alternatives used, and how can translation

alternatives be pruned to maximize CLIR effectiveness?

1.3 Contributions

This dissertation makes the following contributions:

1. A theoretical framework for CLIR, that is, the meaning matching model. The model is developed based on the first principle of IR that the goal is to match what the searchers mean with what the document author meant. In this model, sets of synonymous terms (or synsets) are used as the computational representations of meanings. Statistical term-to-term translations are converted into statistical term-to-synset mappings by conflating synonymous translations into synsets based on synonymy knowledge learned from resources such as WordNet or parallel texts. Experiment results revealed that meaning matching that used bidirectional translation knowledge and synonymy knowledge could outperform the probabilistic structured query method significantly, achieving CLIR effectiveness comparable to monolingual effectiveness.

2. A new technique for using document translation knowledge, that is, the probabilistic document translation method. The technique uses translation probability from document terms to query terms in the estimation of query term weight in a similar way as the structured query method. Experiment results showed that it could be just at least as effectively as the probabilistic structured query method.

3. A new method to establish a fair monolingual baseline. Multiple translations used for CLIR have an expansion effect. If the same translation resources can be used in query/document expansion in monolingual retrieval, then the monolingual effectiveness obtained is a fair upper bound. In this study, statistical synonyms identified from the same parallel corpus were used for monolingual query expansion, and we were able to compare our CLIR effectiveness to monolingual effectiveness achieved in this way.

4. Reference implementation for sharing with other interested researchers to explore a wider range of issues in IR and other related fields.

1.4 Organization of the Dissertation

The dissertation is organized as follows. Chapter 2 reviews the problems of and different approaches to CLIR. CLIR issues and techniques are described from three perspectives: (1) which terms should be translated? (2) how to obtain translation knowledge? and (3) how should that translation knowledge be used?

Chapter 3 introduces the meaning matching model. It describes techniques for acquiring bidirectional translation knowledge and synonymy knowledge, and combing these two sources of knowledge in the framework of meaning matching. In addition, it illustrates that the probabilistic structured query method and its analog, the probabilistic document translation method, are special cases of the meaning matching model.

Chapter 4 describes the design of an English-French CLIR experiment and presents the results. This includes selection of the test collections, training statistical translation models, deriving statistical synonyms, query/document processing, development of the IR engine used in the study, and comparisons of CLIR effectiveness under different conditions.

Chapter 5 describes two sets of experiments that address CLIR with English queries and Chinese documents in a similar way to Chapter 4.

Finally, Chapter 6 concludes the thesis with the main findings, limitations, and implications for future work.

Chapter 2

Cross-Language Information Retrieval: Overview

Research in the field of CLIR has surged during the last decade, as marked by four major IR/CLIR evaluation workshops: the Text Retrieval Conference (TREC), the Cross-Language Evaluation Forum (CLEF), the NII-NACSIS Test Collection for IR Systems (NTCIR) evaluation, and the Topic Detection and Tracking (TDT). Proceedings of these workshops, proceedings of the annual ACM SIGIR Conference on Research and Development in Information Retrieval, and several journals are the major publications of CLIR.

In this chapter, we first introduce the general field of IR in Section 2.1. We present the definition of IR, explain its key concepts and components, and outline the process in which IR operates. We then continue by identifying three main problems unique to CLIR in Section 2.2. Section 2.3 then surveys the main techniques developed to deal these problems. Evaluation of CLIR is discussed in Section 2.4. The chapter concludes with a summary of several CLIR problems and techniques that motivated the study in Section 2.5.

2.1 Introduction to Information Retrieval

This section describes the major processes, concepts, and models of information retrieval.

2.1.1 Several Key Concepts of Information Retrieval

At its highest level, this dissertation concerns Information Retrieval (IR), the task of finding relevant items from a repository of information to solve information problems encountered by users. A typical IR system stores and represents collections of information items. Upon receipt of a user's search request, it forms the search request into a query, initiates a mechanism that matches the query against the representation of each information object in the repository, and nominates a subset of information objects potentially relevant to the user's information need. Here, several concepts are important not only for defining IR but also for clarifying the scope of this study.

The information repository of an IR system consists of a finite set of *information items.* Information items encompass human knowledge and describe events, facts, opinions, etc. They could exist in a variety of forms such as handwriting text, printed text, electronic text, images, music, audio, and video. Usually the retrieval of many non-text information objects can be converted into the retrieval of text, so *text retrieval* has been a key component of modern information retrieval systems. Information objects in text format are usually called *documents.* A set of documents comprises a *document collection.* In this study, we focus on retrieval of documents written in a language different from the language in which the information request is expressed, that is, the task of Cross-Language Information Retrieval (CLIR).

Users come to an IR system because they have *information problems*. Information problems (for example, "I need journal articles on word sense disambiguation.") differ from other kinds of problems (for example, "I'm very hungry.") in that information is needed in order to solve the problems. Information problems create *information needs*. However, clarifying information needs and expressing them accurately is not an easy task, mainly because: (1) users may not be sure what their information problems are and what information they need exactly, especially in the early stage, (2) information needs may change as users interact with IR systems, (3) languages may be inadequate for describing information needs precisely, and (4) IR systems may have constraints on how users' information needs can be expressed and interpreted. In other words, an information need is dynamic and subject to change in the process of IR.

Relevance is perhaps the most important and controversial concept in the field of IR. A document's relevance to an information need could be affected by many factors such as its subject content, novelty, authority, credibility, availability, etc. In this study, we focus on the retrieval of documents that are topically relevant to users' information needs. Such a simplification makes it possible to produce relevance judgments that specify whether a document is relevant to an information request. This is important because relevance judgments can be used as a common-ground for comparing different retrieval systems and evaluating new techniques.

The success of information retrieval is ultimately assessed by the degree of *user* *satisfaction* with the quality of the retrieved information in solving their problems. In a user-centered paradigm, user satisfaction can be studied through questionnaires, surveys, direct observations, etc. In a system-centered paradigm, as is the case of this study, user satisfaction is measured mainly by *retrieval effectiveness*. Two basic quantitative measures of retrieval effectiveness are widely used to derive other effectiveness measures: recall and precision. *Recall* is the proportion of retrieved relevant documents to the total relevant documents, while *precision* is the proportion of retrieved relevant documents to the total retrieved documents. Recall reflects the system's power of including all possible relevant documents, while precision measures the system's capability of excluding non-relevant items. They complement each other, but neither can give a complete assessment of the system's performance alone. They form the basis for system-centered IR evaluation, and many effectiveness measures are derived from them. We will discuss CLIR evaluation in detail in Section 2.4.

2.1.2 Basic Processes of Information Retrieval

An IR system basically has to support three processes: representation of users' information needs, representation of the documents the system contains, and matching the two representations (see Figure 2.1).

The process of representing documents is indexing. Usually, document features are compared, such as the statistical distributions of the content-bearing words within each document. Document features are stored and organized in the docu-

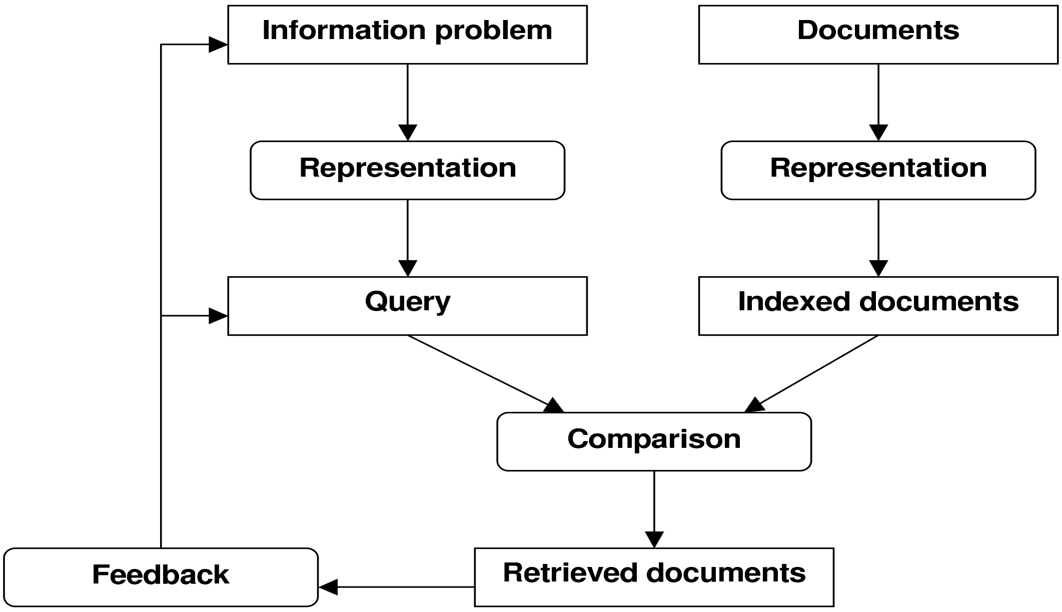


Figure 2.1: Three basic processes of information retrieval [14].

ment index to facilitate matching documents with queries represented in a similar fashion. Indexing usually consists of sub-processes such as analysis of spelling variants, stemming, stopword removal, and term counting. Most IR systems treat each document as a "bag of terms," so only meaning-bearing linguistic units are extracted and used, while other information such as sentence structure is discarded. In this dissertation, we use *terms* to refer to the meaning-bearing linguistic units used to index documents. A term could be a word, a phrase, a synonym set, or a character n-gram. There are generally two types of IR, according to the nature of the terms used to represent queries and documents. Controlled vocabulary IR allows only the use of terms pre-defined in a thesaurus, while free-text IR allows the use of any terms appearing in the document collection. Controlled vocabulary IR requires both the indexer and the searcher to be familiar with the thesaurus terms and to assign consistently same thesaurus terms to the same concepts. When used by experienced indexers and searchers, controlled vocabulary IR can be effective. On the other hand, controlled vocabulary usually requires manual or semi-automatic indexing, which could be impractical when the amount of information that needs to be processed becomes large. In addition, for users who are unfamiliar with the subject domain and the thesaurus, controlled vocabulary can be more of an obstacle than a useful tool. This study focuses on free-text retrieval.

The process of representing the user information need is called *query formulation*. Query formulation takes the information need as its input and produces a query in a format acceptable to the system. Defining the information need, however, can be a complicated process, as suggested by Taylor [82]. However, query formulation in this study is simplified. Given a search topic that expresses user's information request, we automatically extract all the words and treat them as a bag of words for that query.

The matching process compares the query representation with the document representation to produce a subset of documents that are potentially relevant to the information need. The mechanism of matching queries and documents is often referred to as *retrieval model*. Generally, two kinds of matching mechanisms exist, Boolean retrieval and ranked retrieval (described below). Consequently, the result of query/document matching is either a fixed set of documents or a ranked list of documents that are hopefully relevant to the information need.

Ideally, IR systems should retrieve only relevant documents. Due to the uncertainty embedded in the process of representing users' information needs and document content, it is almost inevitable that some relevant documents will be missed while some non-relevant documents will be returned. In addition, information needs may change as users gain a better understanding of the system, the document collections, and the way that the system retrieves and presents documents. Therefore, IR systems usually provide some interaction mechanisms through which users examine and select documents returned by the systems and refine their queries for searching more relevant documents. This is where feedback (as shown in Figure 2.1) happens.

These three basic processes are general to all kinds of IR, although for specific IR some of these processes could be more complicated. For example, in CLIR, queries and documents are written in different languages, meaning either queries and/or documents need to be translated. In this case, query/document translation becomes part of representation. In an interactive CLIR system, users may be also involved in the process of query/document translation, so feedback could be more complicated [28].

2.1.3 Information Retrieval Models

Information retrieval generally falls into two categories: exact match and ranked retrieval. In a system based on exact match, a binary decision is made regarding the relevance of a document to a query. Boolean model is a typical exact match model. There are many retrieval models based on ranked retrieval. The mostly widely used ones includes vector space model, probabilistic model, and models that uses language modeling approaches. We give a brief description of these models in this section.

2.1.3.1 Boolean Model

The Boolean model is based on set theory and Boolean algebra. In a Boolean retrieval system, documents are represented by a set of index terms, but the statistics such as term frequency (TF) and document frequency (DF) are ignored. Therefore, the retrieval in a Boolean system is based on binary decision (that is, a document is either relevant or non-relevant). This is a major difference between Boolean retrieval systems and ranked retrieval systems. In addition, queries in a Boolean system are constructed using Boolean operators such as AND, OR, NOT to connect query terms. A weakness of Boolean query formulation is that information needs may not easily be converted into Boolean expressions. Nevertheless, Boolean retrieval model is still used by many commercial information retrieval systems.

2.1.3.2 Vector Space Model

Vector space model (VSM) shares a common feature with other types of ranked retrieval models: it ranks documents in decreasing order of some measures that corresponds to the relevance of each document to the query. In a VSM, both queries and documents are represented by n dimensional vectors [76]. Each element in a vector is the weight of the corresponding term in the query or document that the vector represents. The relevance score, more precisely Retrieval Status Value (RSV), of a document with respect to a query in a VSM is estimated by the similarity between the query vector and the document vector. Mathematically, this can be realized by the inner product of the query vector and the document vector.

Suppose the query vector is $\vec{q} = (w_{1,q}, w_{2,q}, \ldots, w_{n,q})$ and the document vector is $\vec{d_k} = (w_{1,k}, w_{2,k}, \ldots, w_{n,k})$, we have:

$$RSV(q, d_k) = \vec{q} \cdot \vec{d_k} = \sum_{i=1}^{n} w_{i,q} \times w_{i,k} \tag{2.1}$$

An important problem with this computation is that longer document tend to receive higher RSV due to longer vector length, while longer documents may not necessarily more relevant than shorter documents. A better way to handle the problem is to apply vector length normalization. Mathematically, the inner product of two length-normalized vectors equals to the cosine of the angel between the two vectors, so RSV based on vector length normalization is often referred to as cosine similarity:

$$RSV(q, d_k) = cos(\vec{q}, \vec{d_k}) = \frac{\vec{q} \cdot \vec{d_k}}{|\vec{q}| \times |\vec{d_k}|} = \frac{\sum_{i=1}^{n} w_{i,q} \times w_{i,k}}{\sqrt{\sum_{i=1}^{n} w_{i,q} \times \sum_{i=1}^{n} w_{i,k}}} \tag{2.2}$$

Although there exist a variety of ways to compute term weights, they often utilize two weighting factors: term frequency (TF) and document frequency (DF). TF is the number of occurrences of a term in a document, while DF is the number of documents in which a term occurs. TF is a measure of aboutness, which has beneficial effects on both precision and recall. DF is a measure of specificity, and its principal effect is on precision. In practice, both TF and DF are normalized before they are combined to compute term weight. For example, Salton and Buckley suggested the following way to compute document term weights:

$$w_{i,k} = tf_{i,k}/df_i = \frac{freq_{i,k}}{max_l\ freq_{l,k}} \times \log \frac{N}{n_i} \tag{2.3}$$

where:

- $tf_{i,k}$ is the normalized TF in document d_k;
- df_i is the normalized DF;
- $freq_{i,k}$ is the raw TF in document d_k;
- N is the total number of documents in the collection;
- n_i is the number of documents containing the term (that is, the raw DF).

and the following way to compute query term weights:

$$w_{i,k} = qtf_{i,q}/df_i = (0.5 + \frac{0.5 \times freq_{i,q}}{max_l\ freq_{l,q}}) \times \log \frac{N}{n_i} \tag{2.4}$$

where $qtf_{i,q}$ is the normalized query term frequency and $freq_{i,q}$ is the raw term frequency in query q.

2.1.3.3 Probabilistic Model

The classic probabilistic model was introduced in 1976 by Robertson and Sparck Jones [73]. Given a query q, the probabilistic model attempts to estimate the similarity between a document d_k in the collection and query q in the following way:

$$sim(d_k, q) = \frac{p(R|\vec{d_j})}{p(\overline{R}|\vec{d_j})} \tag{2.5}$$

where $p(R|\vec{d_j})$ denotes the probability that document d_k belongs to the set R of relevant documents and $p(\overline{R}|\vec{d_j})$ denotes the probability that d_k belongs to the set $\overline{R}$ of non-relevant documents. Applying Bayes' rule, the similarity becomes:

$$sim(d_k, q) = \frac{p(\vec{d_j}|R) \times p(R)}{p(\vec{d_j}|\overline{R}) \times p(\overline{R})} \tag{2.6}$$

In this equation, $p(\vec{d_j})|R \times p(R)$ stands for the probability of randomly selecting d_k from relevant set R while $p(\vec{d_j}|\overline{R})$ stands for the probability of randomly selecting d_k from non-relevant set $\overline{R}$. Since $p(R)$ and $p(\overline{R})$ are the same for all documents, they will not affect document ranking. Therefore, the similarity computation can be simplified as:

$$sim(d_k, q) \approx \frac{p(\vec{d_j}|R)}{p(\vec{d_j}|\overline{R})} \tag{2.7}$$

Based on the term independence assumption, it becomes:

$$sim(d_k, q) \approx \frac{\prod_{w_i=1} p(w_i = 1|R)}{\prod_{w_i=1} p(w_i = 1|\overline{R})} \times \frac{\prod_{w_i=0} p(w_i = 0|R)}{\prod_{w_i=0} p(w_i = 0|\overline{R})} \tag{2.8}$$

where:

- $p(w_i = 1|R)$ stands for the probability that term w_i is present in a document randomly selected from the relevant set R;
- $p(w_i = 1|\overline{R})$ stands for the probability that term w_i is present in a document randomly selected from the non-relevant set $\overline{R}$;

- $p(w_i = 0|R)$ stands for the probability that term w_i is absent from a document randomly selected from the relevant set R;

- $p(w_i = 0|R)$ stands for the probability that term w_i is absent from a document randomly selected from the non-relevant set $\overline{R}$.

If we use p_i to represent the first item and q_i to represent the second item, the third item and the fourth item becomes $1 - p_i$ and $1 - q_i$ respectively. Taking logarithm of the product, we finally have:

$$sim(d_k, q) \approx \sum_{t_i \in q} \log \frac{p_i(1 - q_i)}{q_i(1 - p_i)} \tag{2.9}$$

Robertson and Sparck Jones discussed different ways to estimate the parameters and suggested to estimate p_i using r/R and q_i using $(n - r)/(N - R)$ [73], where:

- r is the number of relevant document containing query term t_i;

- R is the total number of relevant documents;

- n is the number of documents containing query term t_i;

- N is the total number of documents in the collection.

When relevance information is not present ($R = r = 0$), a constant factor 0.5 is added. The weight of query term t_i can therefore be estimated as Robertson-Sparck Jones Weight:

$$w_i = \log \frac{N - n + 0.5}{n + 0.5} \tag{2.10}$$

Compared to VSM, this model only considers document frequency. Robertson et al later extended the model based on a 2-poisson distribution to include term frequency information in the computation of similarity scores. They developed a family of weighting functions that are generally known as Best Matching (BM). The most widely used one is BM25 weighting [74]:

$$RSV(q, d_k) = \sum_{t_i \in q} [\log \frac{(N - n + 0.5)}{(n + 0.5)}][\frac{(k_1 + 1)tf}{(k_1(1 - b) + b\frac{dl}{avdl} + tf)} \frac{((k_3 + 1)qtf)}{(k_3 + qtf)}] \tag{2.11}$$

where:

- $RSV(q, d_k)$ is the retrieval status value of document d_k concerning query q;
- N is the total number of documents in the collection;
- n is the number of documents containing t_i;
- tf is the term frequency of t_i in d_k;
- dl is the length of document d_k;
- $avdl$ is the average document length in the collection.
- qtf is the term frequency of t_i in query q;
- k_1, b, and k_3 are parameters in BM25, usually 1.2, 0.75, and 7-1000 respectively.

2.1.3.4 Information Retrieval based on Language Models

Language modeling approaches to IR have been widely studied recently [69, 29, 54, 4, 85, 43, 42]. The underlying idea is that each document in the collection defines a document model M_D, and the document is ranked by the probability that the query was generated by the document model: $P(Q|M_D)$ (query likelihood). The word independence assumption is used so the probability of a unigram document model generating a query could be computed through multiplying the probability of the document model generating each query word or token:

$$P(Q|M_D) = P(q_1, q_2, \ldots, q_n|M_D) = \prod_{i=1}^{n} P(q_i|M_D) \tag{2.12}$$

Alternatively, a unigram language model M_Q for the query Q can be defined. In this case, documents are ranked through estimating the probability of each document D in the collection being sampled randomly from this language model (document likelihood):

$$P(D|M_Q) = P(w_1, w_2, \ldots, w_n|M_Q) = \prod_{w \in D} P(w|M_Q) \tag{2.13}$$

Both models have advantages and disadvantages. The query likelihood model does not have the notion of relevance, and it is difficult to incorporate user feedback and expansion. The document likelihood model does not consider document length, and documents containing frequent words tend to be favored. For these reasons, approaches combining the two models have been suggested [40]. First, a query model M_Q and a document model M_D are estimated. Then, the similarity between the two models is computed based on similarity measures such as cross-entropy:

$$H(M_Q \parallel M_D) = -\sum_{w} P(w|M_Q) \log P(w|M_D) \tag{2.14}$$

Regardless of methods, the key to language modeling approaches is estimating the language model M_D and/or M_Q from D and/or Q. The problem becomes the computation of unigram frequencies through maximum-likelihood estimation and handling of zero frequency words (using smoothing). Among all smoothing techniques, interpolation using background probabilities seems to succeed for such purposes. Work by Miller, et al illustrated this when they developed a two-state hidden Markov model (HMM) to simulate the process of query generation, assuming word independence [54]. This model estimates the probability of a document generating a query term by the normalized frequency of the term, augmented by the unigram frequency of the term in a large corpus in query language. Mayfield, et al integrated this model into their HAIRCUT retrieval system [50].

We have to point out that all these models were original developed to handle monolingual IR, and all of them requires some sort of term matching. When applied to CLIR, since queries and documents are in different languages, directly matching queries and documents will fail. With vector space models and probabilistic models, it is common to see that either queries or documents are first translated into the other language so that CLIR is converted into monolingual IR. With language modeling approaches, a document translation model is often integrated into the computation of unigram frequency. For example, Xu and Weischedel later extended the model to CLIR by integrating a translation model into it [85]. The same model was used by Larkey and Connell in their TREC experiments [41]. As we explain in detail in Chapter 3, we developed a new way of estimating TF and DF of query terms in documents in this study even if they are in different languages.

2.2 Major Issues of Cross-Language Information Retrieval

As a special case of IR, CLIR must consider both issues common to general IR and those unique to CLIR. In this section, we will focus on issues unique to CLIR. We are particularly interested in knowing what these issues are, how they affect CLIR effectiveness, what techniques have been developed already to solve these issues, and what problems remain. CLIR differs fundamentally from monolingual IR by having queries and documents written in different languages. Due to this unique feature, a direct match of query representation and document representation usually will fail. Therefore, either queries or documents, or sometimes both, need be translated before a matching mechanism can be applied. Many issues unique to CLIR can be attributed to the operation of query/document translation.

Issues specific to CLIR have been well recognized. Oard and Diekema addressed three challenges in dictionary-based CLIR: what to translate, where to obtain translation knowledge, and how to use the translation knowledge [60]. In conducting CLIR, one has to first decide the linguistic units to be translated. What to translate usually relates to the available translation resources and language processing tools. The translation resource is probably the most important resource for CLIR, because it influences the way other issues are handled and the overall retrieval effectiveness. The availability of translation resources varies greatly among different languages. The translation resources also plays an important role for people to decide what knowledge to extract and how to obtain such knowledge. The type of the translation knowledge in turn affects how it will be used.

2.2.1 What to Translate?

Translation units can be word stems, words, character N-grams, or phrases. The available translation resource plays an important role in the selection of translation units, and a balanced consideration between translation ambiguity and term coverage is equally important. Machine translation systems use many linguistic features, and are usually capable of translating individual words, phrases, or sentences. Bilingual dictionaries focus on word translation and sometimes phrase translation. Translation models learned from parallel corpora are also primarily intended for word translation, with recent studies beginning to address phrase translation [66]. For a given resource, accuracy increases while coverage decreases when moving from word stems to words, phrases, and, finally, sentences. In other words, high translation accuracy (or low translation ambiguity) often occurs at the cost of low translation coverage. For example, phrases are often less ambiguous than their component words, but a bilingual dictionary with poor phrase coverage will fail to translate most phrases. A dictionary is more likely to cover single words, although some translations might be inaccurate in any particular context due to word sense ambiguity.

Basic natural language processing (NLP) operations, such as word segmentation, stemming, and stop-word removal, are often necessary in the selecting translation units. Word segmentation is simple for languages like English as white spaces between words in written text provide a natural boundary. However, it could be a complicated for other languages such as Chinese, which does not exhibit any explicit word boundary. Adding to the complexity is the issue of segmentation ambiguity, in which sentences may be segmented into different words with different meanings. An early study of dictionary-based query translation for Chinese-English CLIR showed incorrect segmentation often led to rare Chinese words that had translations with low document frequency [62]. As a result, many non-relevant documents were retrieved, and CLIR effectiveness was significantly degraded. The issue of word segmentation is not trivial. It also exists in other languages such as German and Finnish, which have a lot of compound words that may need to be decomposed into component words.

Stemming identifies stems shared by morphological variants. In monolingual IR, stemming reduces the size of the document index and improves recall. Stemming is also useful for CLIR, since it helps to translate morphological variants that may not be covered by bilingual dictionaries. Despite the usefulness of stemming in translation and indexing, it can be difficult to design good stemmers, especially for languages with complicated morphology (e.g., Arabic). In addition, stemming "depth" could be critical since too "light" stemming will have little effect while too "aggressive" stemming may incorrectly pick up words that do not share the same meanings.

Finally, function words and words appearing in many documents are of little importance for IR and should be removed. These function words and frequent words are called *stop-words*. Stop-word removal can reduce the size of the document index. For CLIR based on word translation, pre-translation stop-word removal is important since stop-words can sometimes be translated incorrectly into words with high term weight [46].

2.2.2 How to Obtain the Translation Knowledge?

Translation knowledge may include a broad range of things useful for query/document translation. When translation happens on a word-to-word basis, translation knowledge could mean that for a given term in one language, all the terms in the other language that the term translates to and/or the probability that the term translates to each of them. In this dissertation, translation knowledge means both the multiple translation alternatives and their probabilities. Obtaining translation knowledge requires two steps: acquiring translation resources that contain the translation knowledge and extracting the translation knowledge from the resources. Among the possible translation resources, humans can produce the most accurate and fluent translation. In some interactive CLIR systems, human translators (usually also the information searchers) are involved in selecting translations nominated automatically by machines [28, 21]. However, due to their limited availability, high expense, and low speed, human translators are impractical for any automatic CLIR system. Therefore, translation resources such as bilingual dictionaries, parallel/comparable corpora, and MT systems, are more often used for CLIR.

Each type of translation resource has its strengths and weaknesses in terms of availability, vocabulary coverage, translation ambiguity, and the amount of effort required to develop/acquire it and to extract translation knowledge from it. In deciding which translation resource(s) to use, all these factors should be considered carefully. MT systems usually produce more accurate translations than other translation methods, but building MT systems requires more language resources and human effort. Bilingual dictionaries cover more language pairs than other translation resources. However, machine readable bilingual dictionaries have varied quality of term coverage and usually do not contain translation probability proved useful for CLIR [15]. Sentence-aligned parallel corpora contain pairs of sentences in two languages that are accurate translations of each other. They can used for deriving term-to-term translation models, which have been shown useful for CLIR [85, 39]. However, sentence-aligned parallel texts do not exist for most language pairs, and the translation models derived in this way are domain-specific. Comparable corpora, which are bilingual texts on the same subject, are easier to obtain. Extracting accurate translation knowledge from comparable corpora is a more challenging task. Section 2.3.2 will describe major techniques for obtaining translation knowledge from different resources.

Regardless of the translation resources, often terms cannot be translated as they are not covered by the translation resources. These terms are known generally as Out-Of-Vocabulary (OOV) terms. Since OOV terms are often named entities such as person names, organization names, and geographical locations, failing to translate them can degrade CLIR effectiveness significantly.

2.2.3 How to Use the Translation Knowledge?

The use of translation knowledge is usually studied from two interrelated perspectives: translation disambiguation and weighting translation alternatives. When multiple translation alternatives are available, whether dictionaries or corpora, one has to resolve translation ambiguity (i.e., decide which translation alternative or alternatives to use). Translation ambiguity occurs because of the existence of polysemy, when terms may have more than one meaning. For each polysemous term, bilingual dictionaries usually contain translations that have different meanings. A simple dictionary reference will bring in all these translations. However, in a given context (for example, in a specific query submitted by a searcher) a term usually has just one meaning. Including translations that do not encode the intended meaning in the translated query will lead to retrieval of irrelevant documents. Ambiguity also exists in translations derived from parallel corpora because every term in one language has some chance of translating to every term in the other language.

How severely can translation ambiguity and the failure to translate OOV terms affect CLIR effectiveness? An early study showed that, without translation disambiguation and OOV recovery/compensation, CLIR could only achieve 40-60% retrieval effectiveness of its monolingual counterpart [17]. Most researchers attributed the drop of CLIR effectiveness to failing to translate multi-word expressions as a single unit, including incorrect translations, and failing to translate OOV terms. In the

Mandarin English Information (MEI) project at the 2000 Johns Hopkins Summer Workshop, we investigated the influence of translation ambiguity and incomplete translation on CLIR effectiveness [53]. By manually adding missed translations and removing extraneous translations from dictionary translations, we found that translation ambiguity (inclusion of some inappropriate translations) and incomplete translation (failure to translate some source terms or failure to include some appropriate translations) each accounts for about 30% decrease of retrieval effectiveness, and they accounted jointly for about 40% decrease of retrieval effectiveness, both measured by mean average precision. Although the experiment had its limitations, the results showed the importance of resolving translation ambiguity and OOV terms in CLIR.

The three issues are often interrelated to each other. Development and acquisition of translation resources is the most fundamental, since it largely affects which translation units to select and how to use the translation knowledge. For example, if a bilingual word list is used, translation disambiguation will require extra resources such as a parallel/comparable/monolingual corpus or linguistic tools such as a part-of-speech (POS) tagger. If the translation resource is a statistical word-to-word translation model, both translation disambiguation and weighting translation alternatives can be done on translation probability. In addition, phrase identification may reduce the effort required to do translation disambiguation, while stemming could compensate for the limited coverage of translation resources. Finally, translation disambiguation and weighting translation alternatives are closely related to each other. Good translation disambiguation can minimize, or even eliminate, the necessity of weighting translation alternatives, while a well-designed weighting schema can accommodate translation ambiguity.

Among these issues, translation ambiguity and weighting translation alternatives are common to all languages. For this reason, they have attracted the most intensive research interest in CLIR. Extraction of translation units, alternatively, is a language-dependent issue. In next section, we will review the state-of-the-art techniques designed to address these issues.

2.3 CLIR Techniques: the State-of-the-Art

Over the past decade, a variety of models, approaches, and techniques have been proposed for and applied to CLIR. While some have been commonly accepted, others continue to be improved. In this section, we review the state-of-the-art technology in the field of CLIR.

2.3.1 Pre-Translation Segmentation

Pre-translation segmentation seeks to answer the question of "what to translate," identifying and extracting the linguistic units to be translated. In order to extract the appropriate forms of linguistic units ready to be relayed to the translation routine, several stages of text processing should be performed. Among them, the most important for CLIR include tokenization, stemming, phrase identification, and stop-word removal. Stop-word removal is simple, and we described it in Section 2.2.1. We focus now on techniques for tokenization, stemming, and phrase identification.

2.3.1.1 Tokenization

Tokenization is the process of recognizing words. This usually includes isolating words from each other (word segmentation) and from punctuation marks. It also recognizes abbreviations and acronyms, corrects possible word splits across lines, reduces words to lower case, and removes accents. Most of these operations are relatively simple. Word segmentation sometimes requires sophisticated computer algorithms and language resources. For languages like English, in which white space indicates explicit word boundaries, word segmentation is simple. But for languages like Chinese, in which words in a sentence are concatenated one after another, word segmentation is complicated. It is often an ambiguous process because the same sentences may be segmented in different ways. Such ambiguity limits the accuracy of simple dictionary-based techniques.

Usually, three sources of evidence for automatic word segmentation exist: lexical representations such as a list of known items, algorithmic knowledge such as a heuristic preference for the longest strings, and statistical evidence acquired from a representative collection of text. Each source of evidence has advantages and disadvantages, and practical segmentation schemes exploit multiple sources of evidence. For example, the simplest commonly implemented approach uses a greedy left-to-right search for the longest matching substring in a term list. However, lexicon-based segmentation generally fails when terms not covered by the lexicon are encountered. Statistical segmentation approaches can help overcome this problem and can also help improve the selection among alternative segmentations. However, the accuracy of statistical approaches depends on how closely the corpus used to derive the statistics represents the text to be segmented. Approaches based on syntactic parsing have also been proposed. Since they are computationally expensive, such approaches are rarely used in IR. Several studies also used multiple segmentation hypotheses [62, 85], but using multiple segmentations may produce more incorrect translations than using one-best segmentations.

For other languages, such as German, tokenization also includes decompounding, in which compound words are split into component words. Decompounding is a special kind of word segmentation. Therefore, techniques developed for word segmentation can also decompound with little modification. For example, the simplest way of decompounding uses a greedy left-to-right search for the longest matching substring in a term list. Statistical approaches can refine or select results of dictionary-based decompounding. Chen developed a German/Dutch decompounding technique that uses both lexical knowledge and statistical knowledge [10]. Compounds were decomposed first with a dictionary that contained only single words (compounds were excluded). If different ways to decompose a compound existed, the one with the minimal number of component words was chosen. Furthermore, if more than one decomposition with the minimal number of component words occurs, the one with the highest probability was selected. The probability of a decomposition was computed by multiplying the probability of each component word, which was defined as its relative frequency in a large corpus. Experiments in both monolingual IR and CLIR showed that decomposition with this technique improved retrieval effectiveness. Monz focused on decompounding German noun-noun compounds with a German dictionary and a German POS tagger [56]. His monolingual retrieval experiments also showed the positive effects of decompounding.

2.3.1.2 Stemming

In CLIR, stemming can be performed either before or after query/document translation. Just like stemming in monolingual IR, post-translation stemming in CLIR attempts to match words that share the same stems (hence hopefully share the same meanings). The main reason for pre-translation stemming is to recover OOV terms. Basically, when the direct match of words to translate and words contained in the translation resource fails, matching their stems may succeed, allowing translation to succeed. Many rule-based stemming algorithms have been suggested for English, including the widely used Porter stemmer [70]. While accurate, rule-based stemmers have disadvantages. Since morphological rules vary greatly between languages, the portability of rule-based stemmers from one language to another is usually poor. For the same reason, development of a rule-based stemmer for a new language could be expensive.

Stemmers based on statistical techniques, alternatively, can overcome these problems. In TREC-3, Buckley et al demonstrated a simple stemmer for Spanish could be obtained easily by examining lexicographically similar words to discover common suffixes [8]. Using a statistical rule-induction technique, Oard et al developed statistical stemmers for French, German, and Italian based on a similar idea of using corpus statistics to find common suffixes for these languages [61]. Their experiments showed stemmers developed in this way could produce improvement for French, but no improvement was found for German or Italian.

2.3.1.3 Phrase Identification

While the use of automatically identified phrases in indexing and searching in monolingual IR has led to small or inconsistent improvement of retrieval effectiveness over the use of pure words, phrase translation in CLIR can lead consistently to better retrieval effectiveness. Phrase dictionaries can be used to identify phrases from text. If no phrase dictionary is available, corpus-based techniques can be used. There are basically three ways to identify phrases: statistical recognition (for example, [23]), POS tagging (for example, [3]), and syntactic parsing (for example, [22]). Statistical recognition exploits term co-occurrence information. For example, to identify two-word phrases, a corpus first can be segmented into overlapping word bigrams, which in turn are ranked in decreasing order by some combination of their term frequency and inverse document frequency. Those receiving high ranks are recognized as phrases. Another way to recognize phrases is to assign automatically POS tags to each word in a text using a probabilistic or rule-based POS tagger. Phrases then can be recognized by POS patterns of consecutive words, for example, two adjacent nouns or an adjective followed by a noun form good phrase candidates. Finally, syntactic parsers can also be used to perform phrase identification. Syntactic parsing of input text creates syntactic structures such as noun phrases which can be regarded as phrases.

2.3.2 Translation Knowledge Acquisition

Translation knowledge acquisition is achieved through two steps: acquiring translation resource and extracting translation equivalents from the translation resource. We focus on two translation resources, bilingual dictionaries and bilingual corpora, since they are the most important and widely-used translation resources.

2.3.2.1 Using Bilingual Dictionaries

The simplest way to acquire bilingual dictionaries is to search the Web. Universities, research institutes, and non-profit organizations make language resources available to interested researchers. However, dictionaries found this way often have limited vocabulary coverage, and CLIR based on translations from these dictionaries can have low effectiveness. Furthermore, such resources cover only a few languages. For most languages, downloadable bilingual dictionaries simply are not available.

Another potential translation resource involves printed bilingual dictionaries. Printed dictionaries contain multiple pages, and each page contains multiple entries. Different entries have regular and repeated structure. For example, each entry is usually distinguished by font properties (bold, italic, size, etc.) or layout features (indentations, bullets, etc.). Ma, et al developed an automated, but user guided, approach to parameterize and learn the physical structure of the dictionary page and the semantics of the dictionary entries [49]. However, CLIR with queries translated using an English-French dictionary acquired with their approach performed significantly worse than using a bilingual term list downloaded from the Web. This indicated these techniques need improvement before bilingual dictionaries acquired using them could be effectively applied to CLIR.

2.3.2.2 Using Bilingual Corpora

For some high density languages, parallel corpora can be acquired from a variety of sources. For example, international organizations such as the United Nations (UN) has many rules, regulations, etc. published in several languages. The European Union publishes official documents in several European languages. Documents from the Canadian Parliament (HANSARD corpora) are available in both English and French. Hong Kong News and Hong Kong Laws are available in Chinese and English. Finally, the Bible has been published in many languages, which could be an important resource for low density languages, for which other types of parallel corpora are not readily available.

The Web is also a potential source of parallel/comparable corpora. Work by Resnik [72] and Nie [58] used simple, but seemingly effective, techniques to find parallel Web pages (that is, Web pages that are translations of each other). In these studies, simple heuristics such as anchor text and HTML structure were used to select candidate Web sites. With the selected parallel Web sites in English and French, Nie derived translations at the word level with the statistical Machine Translation (MT) technique described below. A comparison of CLIR with this model and a similar model built with the Hansard parallel corpus indicated they achieved comparable effectiveness. Resnik, et al produced a bilingual term list with parallel Web pages collected in a similar fashion [65]. Query-translation with the term list also led to CLIR effectiveness that was comparable to that with a term list downloaded from the Web. Both studies showed mining the Web for parallel documents a feasible, promising way to acquire corpus-based translation resources for CLIR.

Other studies such as [85] also used translation knowledge learned from aligned parallel corpora using statistical MT techniques. Different from rule-based MT, which requires human-written linguistic rules (syntax, grammars, etc.) for the source language and the target language, statistical MT is a data-driven technique that seeks to automatically "learn" translation knowledge from bilingual corpora. Based on the noisy channel model [79], IBM developed a series of models with increasing complexity [6]. The basic idea of statistical MT is as follows: given a sentence f in one language, the system's task is to find a sentence e in another language such that the probability $p(e|f)$ is maximized. The probability can be estimated by multiplying a priori probability $p(e)$ and a conditional probability $p(f|e)$ by applying Bayes rule. That is:

$$argmax_e \; p(e|f) = argmax_e \; p(e)p(f|e) \tag{2.15}$$

The problem is decomposed into two subproblems: estimating $p(f|e)$ and $p(e)$ respectively. $p(e)$ is a langauge model that specifies the probability that sentence e is generated. $p(f|e)$ is a channel model (or translation model) that defines the probability of sentence e could have "become" the observed sentence f after passing through the noisy channel. The advantage is that $p(e)$ models syntactical constraints, while $p(f|e)$ models the lexical aspects. The language model $p(e)$ can often be described with N-gram models. The estimation of the translation model $p(f|e)$ requires sentence-aligned parallel corpora. Basically, if two words co-occur frequently in a parallel sentence pairs, they are likely to be translations of one another. Detailed explanation of statistical MT can be found in [6, 65]. A statistical MT toolkit called GIZA++ has been developed [64]. When running GIZA++, users can choose which IBM models to use and how many iterations to perform for each model. This facilitates the study of the influence of different parameter settings on the accuracy of statistical MT.

Cross-language latent semantic indexing (CL-LSI) by Littman, et al is a different technique of using bilingual corpora for CLIR [48]. CL-LSI uses a parallel training corpus aligned at document level, with translation of each document being adjoined with the document. Singular value decomposition techniques are then used to map sparse term-vectors to a dense semantic space. Each word is represented by a short vector of real numbers giving its position in the reduced semantic space. Distance between two vectors can be calculated and used as a measure of similarity between the two words. Since the training corpus contains terms in both languages, the reduced semantic space will contain terms from both languages. In this semantic space, words that are closely related with one another will have similar representations. Each document in the collections can then be added into the space by using the weighted sum of its constituent words. A query can be represented by a vector in the same way, whether it is in the same language as the documents or in the other language. Thus CLIR can be carried out using vector space retrieval techniques.

Work by Sheridan used a collection of comparable news articles aligned by dates and language-independent subject codes [80]. A German query was first used to search German documents. German documents with higher ranks in the retrieved list were extracted. Italian documents comparable to these German documents were found based on the same dates and subject codes. The important terms extracted from this pool of Italian documents formed an Italian query. Thus, translation of a German query into an Italian query was achieved, although term-to-term translation was left unsolved. In practice, a similarity thesaurus was pre-constructed in a similar fashion. When a new query is entered, the most similar terms in the target language can be extracted directly from the similarity thesaurus. Davis and Dunning exploited a similar idea but with a parallel corpus [16]. In their study, English queries were translated by replacing the original query terms with the 100 most frequent terms in the top 100 retrieved documents from the Spanish side of the parallel corpus. Picchi and Peters developed a method with a comparable corpus and a bilingual lexical database to develop translation equivalents based on word context [67]. However, the effectiveness of their method when applied to CLIR was not reported.

2.3.2.3 Transitive Translation

While resources for direct translation between Language A and Language B may not exist, it is possible that translations exist between Language A and Language P and between Language B and Language P. In this case, translation from Language A to Language B can be achieved through translation from Language A to Language P and then from Language P to Language B. Language B usually is called a "pivot" language, and this technique of translation generally is referred to as transitive translation [1, 24]. One major problem with transitive translation is that it could double the translation errors because two translation is performed twice. Ballesteros managed to improve CLIR effectiveness with transitive translation to 67% of monolingual performance in the target language although this is still worse than the 79% monolingual performance achieved by direct translation [1]. Gollins and Sanderson suggested a "lexical triangulation" approach to decrease errors due to transitive translation [24]. In that study, two pivot languages were used. A source term was translated into the target language twice with transitive translation using two pivot languages respectively. Target translations that appear in both transitive translations are considered better than translations that appear in only one transitive translation. According to their analysis, lexical triangulation has the potential to outperform direct translation.

2.3.3 Translation Disambiguation

Translation ambiguity is well recognized as one of the most important factors that influence CLIR effectiveness. Many techniques have been proposed to resolve this problem. These techniques either use word statistics computed from corpora or explore syntactic constraints or other evidence such as dictionary structure.

2.3.3.1 Exploring Bilingual Dictionaries

In the early days of CLIR, research focused on the selection of “one-best” translation from multiple translation candidates provided by bilingual dictionaries. Since most bilingual dictionaries do not provide translation probabilities, some heuristics or text analysis was performed in to choose the most accurate translation. In an early CLIR studies with dictionary-based query translation, Ballesteros simply selected the first translation from the bilingual dictionary, assuming the most probable translation often is given at the first position. Such an approach has some obvious limitations, being: (1) not all dictionaries list the most probable translation first (some list multiple translations alphabetically), and (2) even if the most probable translation is listed first, no guarantee is given that the most probable translation is the best translation. As a result, selecting the first translation did not produce CLIR effectiveness better than using all translation alternatives.

Given several bilingual dictionaries, the instances of a source term’s translation appearing in these dictionaries can help decide the most probable translation. If a translation appears in all dictionaries, we have good reason to assess it as a good translation, and perhaps as the most possible translation. If the translation appears only in one dictionary, it is probably a rare translation, or perhaps incorrect. Similar intuition has been used by several researchers in merging bilingual dictionaries - translations in several dictionaries are assigned more weight than translations found in fewer dictionaries (for example [15]). The translation triangulation technique described above is based on the same idea.

The most effective technique of translation disambiguation based on dictionaries is phrase translation. Phrases in input text first can be identified with the techniques described in Section 2.3.1. The simplest phrase translation approach uses a bilingual dictionary to perform both phrase identification and translation. Even such a primitive technique can improve retrieval effectiveness. Several studies have demonstrated this [3, 53, 27]. Unfortunately, phrase coverage in bilingual dictionaries is limited. When this happens, how can OOV phrases be translated as phrases instead of individual component words? Corpus-based approaches often can answer this question.

2.3.3.2 Exploring Corpora

Davis and his colleagues conducted early studies on the usefulness of parallel corpora for translation disambiguation [16, 18]. In one study, English queries were tagged first with a statistical POS tagger. For each English query term, a bilingual dictionary was used to select Spanish equivalents that matched its POS. The English query was used to retrieve a set of documents from the English side of a parallel, aligned corpus. Each Spanish translation of an English query term was used to retrieve a set of documents from the Spanish side of the parallel corpus. Davis, et al chose the Spanish translation whose Spanish retrieval result most resembled the English retrieval result. Their experiments with the TREC-5 test collections showed while query translation without disambiguation could only achieve about 50% of monolingual effectiveness, combining POS and corpus-based disambiguation improved CLIR effectiveness to more than 70% of monolingual effectiveness. Contradictory results were obtained when corpus-based disambiguation was used independently - it decreased CLIR effectiveness with the TREC-5 test collection (39% of monolingual effectiveness) while increasing it with the TREC-4 collection (67% of monolingual effectiveness) [17].

Term statistics computed from monolingual corpora may be also used to decide translation likelihood. The simplest technique uses the unigram frequency of each translation candidate in a target corpus, where the translation with the highest unigram frequency is chosen as the most probable translation. While this could assist choosing the most probable translation among a group of synonymous translations, it may be inappropriate for selecting the best translation from polysemous translations. Synonymous translation ambiguity seems to have less influence on CLIR effectiveness than polysemous translations. However, none of the studies differentiated these two types of translation ambiguity.

In a natural extension to translation selection based on unigram frequency, contextual information, such as word co-occurrence statistics in a target corpus, is used. Techniques following this idea hypothesize correct translations of two words tend to co-occur more frequently than incorrect translations. Most studies investigated the effectiveness of translation disambiguation using target language corpora. In a study by Ballesteros and Croft, a POS tagger was used to select target translations with the same POS tags as their source query terms [3]. Translations of each source term form a set. For two source words, all possible two-word sets $\{a, b\}$ were generated such that a is a translation alternative of one source word, and b is a translation alternative of the another source term. Word co-occurrence statistics were then computed for words appearing together within a window of 250 words. Finally, all sets were ranked by their co-occurrence statistics and the one receiving the highest rank was considered the "best" translation. Phrases were first identified with POS tagging and then translated with either the bilingual dictionary or with the co-occurrence model (if they were covered by the dictionary). They found both approaches improved CLIR effectiveness significantly over baseline translation for which no disambiguation was used. Adding co-occurrence phrase translation to dictionary phrase translation led to better effectiveness.

In NTCIR1, Lin, et al also studied word co-occurrence information for translation disambiguation [47]. In their study, mutual information (MI) [11] was used to measure word cohesion of two translations x and y within a text window of 3 for two query terms. However, translation disambiguation using co-occurrence statistics computed this way degraded slightly than selecting the first translation in the bilingual dictionary. They suspected either the small size of the training corpus, or domain differences between the query set and the training corpus.

Gao, et al designed a word co-occurrence model that considers the distance between two words in computing their cohesion score [22]. Experiment with the TREC-9 CLIR test collection showed translation disambiguation with this technique led to significant improvement of performance over translation without disambiguation. However, it is still unclear whether this model outperforms the traditional co-occurrence model that uses a fixed-size text window.

2.3.4 Dealing with Out-Of-Vocabulary Terms

When source terms are not found in the translation resource, they cannot be translated, and hence will not contribute to query-document matching. Unfortunately, OOV terms are often proper names, technical terms, abbreviations, and acronyms, which are important for IR. A study by McNamee and Mayfield [52] and a study by Denmer-Fushman and Oard [20] give insights of the effect of the size and quality of translation resources on CLIR effectiveness. Various techniques have been proposed to solve this problem. OOV terms of abbreviations and acronyms can be resolved to full forms through dictionary reference, if abbreviation/aronym dictionaries are available. We focus on techniques for resolving OOV terms in full forms. The major techniques include transliteration, backoff translation, and pre-translation expansion.

2.3.4.1 Transliteration

Transliteration can be use wither orthographic mapping or phonetic mapping. For languages sharing similar alphabets, orthographic rules specify how certain substrings in one language are spelled in another language. OOV terms then can be transliterated using these rules. For example, Buckley, et al explored the idea in TREC-6 [7]. In their experiment, English words were treated as misspelled French words. Using a spelling error correction technique, the most probable French word that could have led to each English word was found and used as the French translation. Such techniques, although simple, were effective.

However, for languages that do not share any alphabet (such as Chinese and English), orthographic mapping rarely works. In this case, phonetic mapping can be considered. OOV terms in the source language are first converted into their phonetic representations. The phonetic representations can then be mapped into another language using phonetic mapping rules between the two languages. Finally, phonetic representations in the target language are converted into character sequence. Phonetic mapping rules between two languages can be derived using either statistical approaches [37] or linguistic experts. Qu, et al used phonetic transliteration in their Japanese-English CLIR experiments and observed consistent improvement of CLIR effectiveness over experiments without transliteration [71].

2.3.4.2 Backoff Translation

In our discussion of pre-translation stemming, we mentioned stemming of source terms in either the input text or the translation dictionary or both can recover OOV terms partially. Oard, et al developed a "backoff translation" technique to maximize term coverage while limiting the introduction of spurious translations [61]. The technique consists of four stages: (1) match the surface form of an input term to surface forms of source language terms in the translation dictionary; (2) if it fails, match the stem of the input term to surface forms of source language terms in the dictionary; (3) if this still fails, match the surface form of the input term to stems of source terms in the dictionary; (4) if this fails again, match the stem of the input term to stems of source terms in the dictionary. Experiments with the CLEF

2000 collections showed improvement of retrieval effectiveness when using backoff translation.

2.3.4.3 Query Expansion

Another common technique to mitigate the problem of OOV terms involves query expansion. Query expansion can be done before or after translation. In the case of pre-translation query expansion, monolingual retrieval in the query language is performed first with a set of queries on a comparable document collection. The most important terms from top-ranked documents are selected and added to each query to create a new set of queries. The new set is translated into the document language and used to search the target document collection. The rationale for pre-translation query expansion is that it brings in useful terms to be translated into the target language, hence increasing CLIR effectiveness. This is particularly useful for short queries (for example, just two or three terms). Generally, failure to translate one or two terms for a short query will have a much larger impact on CLIR effectiveness than for a long query. Therefore, pre-translation query expansion could be very effective for short queries. Besides comparable document collections, pre-translation expansion terms can be obtained from other resources such as synonym dictionaries.

A similar way of compensating for poor queries is to expand the translated query, which is called post-translation query expansion. As the name indicates, post-translation expansion adds content terms extracted from top-ranked documents retrieved with a translated query to the query to de-emphasize the effect of inappropriate translations. The effect of post-translation query expansion may vary according to the experiment condition. Early experiments by Ballesteros showed post-translation expansion could be effective for long queries but have little effect on short queries [2]. A recent study by McNamee and Mayfield showed pre-translation query expansion consistently improved CLIR effectiveness. Post-translation query expansion was effective for CLIR with poor translation resources, and had little effect with comprehensive translation resources [52].

2.3.4.4 Document Expansion

Similarly, enriching queries with useful terms selected from top-ranked documents can also be applied to documents, that is, document expansion. Document expansion was introduced first for the retrieval of error-prone automatic speech transcription by Singhal, et al [81] and later applied to CLIR by Levow and Oard [46]. Document expansion for CLIR works as follows. First, documents are translated into the query language. Each translated document then is used as a query to search a comparable collection of documents in the query language. Important terms are selected from top-retrieved documents and added to the translated documents, hence an "expanded" document collection is created. This hypothesizes that document expansion could add useful terms missed by automatic document translation that are helpful for improving CLIR effectiveness. The study by Levow and Oard, however, only showed relatively small improvement of CLIR effectiveness when document expansion was used.

2.3.4.5 Using the Web

Finally, the Web represents another useful resource for solving the problem of OOV terms. For example, for a given OOV term, a set of Web documents can be identified with some language identification techniques. Then, the OOV term is referenced to the Web documents. If the pattern of a term in the other language proceeding the OOV term frequently appears in the retrieved documents, the leading term is likely to be the translation of the OOV query term. The idea is based on the observation that it is common to see terms, in particular technical terms, borrowed from another languages are usually kept in their original form right after their translations in the native language [86]. However, it might be only true for languages that use quite different character sets, such as English and Chinese.

2.3.5 Using the Translation Knowledge

Translation disambiguation or translation selection in CLIR may still produce more than one translation for each source term, since more than one translation may be appropriate. However, including all translations in the queries or documents will change term statistics such as TF, DF, and document length. As a result, terms with more translation alternatives could contribute more to query-document matching than terms with fewer alternatives. This is often problematic. Over the years, techniques have been designed to address this issue. These techniques include weighted Boolean model, balanced translation, structured queries, weighted IDF, and probabilistic structured queries.

2.3.5.1 Weighted Boolean Model

Hull developed a weighted Boolean model based on a term independence assumption [32]. If two events A and B are independent, then:

$$P(A\ AND\ B) = P(A \cap B) = P(A) * P(B) \tag{2.16}$$

$$P(A\ OR\ B) = P(A \cup B) = P(A) + P(B) - P(A) * P(B) = 1 - [1 - P(A)] * [1 - P(B)] \tag{2.17}$$

$$P(NOT\ A) = 1 - P(A) \tag{2.18}$$

Let $t_i\,(i = 1, 2, \ldots, n)$ be a query term and d be a document. The following Boolean weighting function can be derived:

$$P(t_1\ AND\ \cdots\ AND\ t_n|d) = \prod_{i=1}^{n} P(t_i|d) \tag{2.19}$$

$$P(t_1\ OR\ \cdots\ OR\ t_n|d) = 1 - \prod_{i=1}^{n} [1 - P(t_i|d)] \tag{2.20}$$

$$P(NOT\ t_i|d) = 1 - P(t_i|d) \tag{2.21}$$

This model can be extended easily to incorporate user-specified importance of each query term by adding a weighting factor to each query term. The model can be applied to manual or automatic retrieval. For manual retrieval, the user can structure his/her query in the same way as in standard Boolean model. For automatic monolingual retrieval, the relationship among query terms is usually considered to be Boolean AND; for automatic CLIR, original query terms can be linked with Boolean conjunctions (*AND* operators), and translation alternatives of each query term can be linked with Boolean disjunctions (*OR* operators). The Boolean conjunction affects disambiguation since "the correct translation equivalents of two or more query terms are much more likely to co-occur in the target language than any incorrect corresponding translation equivalents." [32] The Boolean disjunction can suppress the weight of query terms with many translation equivalents. Therefore, translation disambiguation and weighting occur in the same model without using extra resources such as a corpus.

In Hull's study, the weighted Boolean model performed better than the vector space model in both monolingual IR and CLIR experiments. Furthermore, with the weighted Boolean model, CLIR achieved about 84% effectiveness of monolingual IR. However, since manual query processing (discarding obvious translation errors and adding missing translations, but no disambiguation) was involved, it is unclear how the model could be used in a pure automatic mode. Hull's weighted Boolean model represents one of the early efforts to address the problem of translation ambiguity through term weighting. Other structured query approaches share a similar idea.

2.3.5.2 Balanced Translation

The most intuitive way to mitigate the unbalanced contribution of multiple translation alternatives is to average the term weights. This idea was simultaneously introduced at the third Topic Detection and Tracking evaluation by two teams [44, 46]. Levow and Oard called this idea "balanced translation." in which the weight of a source term is defined as the arithmetic mean of the weight of each of its translations. Term weight in turn can be computed with standard term weighting schemes. With balanced translation, rare translations tend to contribute more to the source term weight than common translations.

2.3.5.3 Structured Queries

Ballesteros and Pirkola simultaneously introduced another technique for computing a source term's TF and DF based on its translations' TF and DF [3, 68]. Since Pirkola studied the technique more intensively, we introduce Pirkola's structured queries. The technique is based on InQuery's synonym operator, which was originally designed to support monolingual retrieval through thesaurus expansion. A set of synonyms ($t_i, \ i = 1, 2, \ldots, n$) designating a concept c - a pseudo-term - are grouped with a synonym operator, and the TF and DF of the pseudo-term in a document d_j are computed in the following way:

$$TF(c, d_j) = \sum_{i=1}^{n} TF(t_i, d_j) \tag{2.22}$$

$$DF(c) = |\bigcup_{i=1}^{n} \{d | t_i \in d\}| \tag{2.23}$$

That is, the TF of the pseudo-term in a document is the sum of the TF of its synonyms in that document, while its DF is the total number of documents that contain at least one of its synonyms.

In Pirkola's experiment, Finnish queries were automatically translated into English through dictionary reference. Multiple translation alternatives of a Finnish query term were treated as synonyms, and InQuery's synonym operator was used to compute the Finnish query term's TF and DF in the English document collection. Pirkola called this technique "structured queries." Inspired by Pirkola's work, several researchers have used this type of structured translation technique in different experiment settings with different language pairs and different test collections, and it has led consistently to improvement of CLIR effectiveness. Some researchers also modified this approach. According to [15], Kwok introduced a variant to Pirkola's method to reduce implementation complexity by replacing the union operator with a summation:

$$DF(c) = \sum_{i=1}^{n} DF(t_i) \tag{2.24}$$

Darwish and Oard suggested another alternative of using the maximum DF among all translation alternatives [15]:

$$DF(c) = MAX_{i=1}^{n}[DF(t_i)] \tag{2.25}$$

Experiments by Darwish and Oard showed no significant difference among Pirkola's method and these two variants [15].

The structured query method is a conservative strategy because if any translation has a high document frequency (common in the collection), the overall document frequency of the query term will be high too. Hence, its term weight will be low. Balanced translation, by contrast, allows uncommon translations to contribute their relatively high term weights to the source term on a more equal basis. Meng, et al used both methods in their English-Chinese CLIR studies [53]. Their results showed balanced translation coupled with post-translation character bigram re-segmentation could outperform the structured query method. In an NTCIR-2 study by Oard and Wang [63], however, structured queries outperformed balanced translation. According to the authors, the conclusions may differ because of query term selection, use of phrase translation, document collections, and the exhaustiveness of relevance judgments in the two studies. Clearly, balanced translation is more sensitive to rare translations than the structured query method. For example, if some query terms in the source language are selected incorrectly due to incorrect word segmentation in Chinese, and they have rare translations, balanced translation will have poor performance.

2.3.5.4 Weighted IDF

All the techniques for weighting translation that we have discussed are designed for the situation in which the knowledge of the probability of a source term being translated into a target term is unavailable. This is often the case with machine-readable dictionaries. In other situations, translation probability may be available. For example, when multiple bilingual dictionaries are merged, the frequency of a translation pair appearing in these dictionaries can be used to estimate its probability. This technique has been used by CLIR researchers for deriving a probabilistic bilingual dictionary from several bilingual dictionaries without translation probability [85]. In addition, it has become common to use translation knowledge derived from bilingual corpora, which always contains translation probabilities. Since translation probability indicates the likelihood of one term translating into another, it is natural to incorporate the probability into the computation of term weight.

In CLEF 2001, Nie designed a technique that used translation probability to modify Inverse Document Frequency (IDF) of multiple translation alternatives [57]. In his study, alternative translations and their probabilities were obtained from a collection of parallel Web documents automatically aligned at sentence level. His experiment results showed CLIR effectiveness improved with the weighted IDF approach for CLEF 2001 queries, while it decreased with CLEF 2000 queries (the document collection in the two experiments is the same). The author tentatively concluded the weighted IDF approach worked better for queries that had more relevant documents in the collection.

The weighted IDF approach is similar to the balanced translation approach. In fact, balanced translation can be viewed as a special case of weighter IDF, for which uniform translation probability is assumed.

2.3.5.5 Probabilistic Structured Query

Darwish and Oard pushed Pirkola's method one step further by integrating translation likelihood in computing TF and DF of the source term in the target document collection [15]:

$$WTF(c, d_j) = \sum_{i=1}^{n} TF(t_i, d_j) * P(t_i|c) \tag{2.26}$$

$$WDF(c) = \sum_{i=1}^{n} DF(t_i) * P(t_i|c) \tag{2.27}$$

where $P(t_i|c)$ is the probability of query term c translating into document term t_i. Darwish and Oard called this family of translation techniques "Probabilistic Structured Queries" (PSQ). They tried weighting only TF (WTF), weighting only DF (WDF), and weighting both (WTF/DF), and compared them with Pirkola's structured query and one-best translation. Their experiment results showed WTF/DF led to the best CLIR effectiveness among all query translation techniques. In addition, WTF/DF was the least sensitive among all techniques to the number of low-probability translations used. Through a series of experiments, Darwish and Oard showed PSQ could significantly outperform structured queries as the number of translation alternatives increased.

2.3.6 Other Techniques

We have reviewed techniques specifically designed for CLIR. Other techniques apply to the general problem of IR and have proved to be effective for CLIR. Two such techniques that are of particular importance to this study are the language modeling approaches to CLIR and using bidirectional translation knowledge.

2.3.6.1 Language Modeling Approaches

Hiemstra showed structured queries can be derived from language model-based IR approaches [30]. He compared the effectiveness of one-best translation, unstructured queries, and structured queries with a small Dutch-English test collection from CLEF. He found the structured query method consistently outperformed one-best translation and unstructured queries and even manual translation disambiguation, but its effectiveness was still worse than monolingual effectiveness (the best structured queries achieved 83% of monolingual effectiveness, as measured by mean average precision). The study had some limitations. For example, only 24 topics were used, which made statistical significance tests less reliable. Dictionary merging was done in an ad hoc way (adding dictionary frequency and corpus frequency), making statistical accuracy questionable in reflecting the true translation probability.

In his dissertation study on language model-based IR, Kraaij compared the structured query method with language modeling approaches with some CLEF test collections that contain queries and documents in English, French, and Italian [39]. He developed CLIR models of probabilistic query translation, document translation, and query translation plus document translation (both being translated to a third language) using language modeling approaches. He found that all three models significantly outperformed the structured query method, achieving at best about 93% of monolingual effectiveness. He noted the relatively poor performance of structured queries was due to (1) the large number of translation alternatives obtained from parallel Web pages, and (2) translation probability was not used. Also of interest, in addition to pruning translations using their probability, Kraaij also tried a cross-entropy criterion for the same purpose. No significant difference was found between these two approaches.

2.3.6.2 Bidirectional Translation

Translation can be done in either direction, from the query language to the document language or the opposite. One way to use translation in both directions merges the ranked list from query translation with the ranked list from document translation. The hypothesis is the merged ranked list may lead to CLIR effectiveness better than either of the individual ranked lists. Ranked lists can be merged by exploiting either the rank or the relevance score. McCarley found merging ranked lists generated using query translation and document translation yielded a statistically significant improvement in mean average precision over that achieved by either approach alone [51]. McCarley demonstrated this effect in a single context: translation probabilities estimated using IBM model 1, with merging based on normalized scores for each document in each list.

The importance of McCarley's study is that it showed the usefulness of combining translation knowledge in both directions. Similar effects of combining a ranked list from query translation with a ranked list from document translation were confirmed by another study that involved the retrieval of Japanese documents and Chinese documents using Korean queries [36]. In that study, bilingual dictionaries were used to translate queries and documents in a word-to-word style, with all translation alternatives being used. Korean-Japanese CLIR experiment results showed merging ranked lists led to at least 10% increase of mean average precision over query translation or document translation alone, while Korean-Chinese experiment results showed at least 20% improvement. A study in CLEF 2001 that used document translation knowledge for query translation disambiguation also showed the usefulness of bidirectional translation knowledge in improving CLIR effectiveness [5].

2.4 CLIR Evaluation

There are two basic types of information retrieval: retrospective retrieval and information filtering. In retrospective retrieval, the document collection is relatively static while search requests are presented to the system on the fly. Retrospective retrieval is fundamental since many other types of retrieval are built upon it. Even for retrospective retrieval, we may be interested in many aspects of the system such as, how fast the system indexes a document collection, how much space it takes to build the index files, how easily the document repository and index can be updated, how well the system helps people define, express, and refine their information needs, how quickly the system responds to users' search requests, how accurately the system returns documents that are relevant to users' information needs, how well the system helps people browse retrieved documents, how effectively a searcher's relevance judgments are used to improve search, how readily the documents retrieved can be used by users, and how easily people can learn to use the system. Each of these aspects corresponds to one or several functions of the system. In reality, it is impossible to develop all functions at one time. We usually divide them into blocks of functions, develop the blocks, and form an integrated system. While it is impossible to determine how well a system will serve its users without all the expected functions being fully developed, it is desirable to assess whether a newly-completed component serves its expected function.

In this study, we do not intend to develop and evaluate a full-fledged retrospective CLIR system that will support all user-system interactions. Instead, we focus on development and evaluation of automatic CLIR techniques that will become an integrated part of operational CLIR systems. In particular, we would like to develop automatic techniques that can rank documents in a foreign language as accurately as possible in response to users' search requests. Retrieval effectiveness is our primary concern.

While previous sections in this chapter focus on the discussion of techniques for CLIR, this section focuses on the evaluation of these techniques. Evaluation methods vary with the nature of the techniques to be evaluated, particularly the available resources and the nature of the intended function. In this study, we used *test collections* (explained below) to evaluate the effectiveness of proposed techniques. IR evaluation with test collections has a relatively long history, starting with the Cranfield experiments [12, 13] and continuing with pioneer researchers such as Salton [75] and Sparck Jones [34]. However, it took until the early 1990's to become the most widely used evaluation method, when the first TREC was held [26].

2.4.1 Test Collections

A test collection consists of a document collection, a set of search topics, and assessments of each document's relevance to each search topic.

2.4.1.1 Document Collections

During the past decade or so, the major IR evaluation workshops, TREC, CLEF, NTCIR, and TDT have produced many document collections particularly in the domain of news. Documents can have multiple fields such as titles, abstracts, keywords, and paragraphs. Each document in a collection is specified by a unique document identification number. In the these major IR evaluation workshops, document structure is represented by SGML markup.

2.4.1.2 Search Topics

Search requests, usually called search topics, are a set of written statements specifying what kind of information is needed and what makes a document relevant to each topic. Search topics are different from queries; they are descriptions of information needs and are independent of the retrieval systems (query formulation, matching algorithms, etc.). Queries must be formulated using search topics and the specific system used to retrieve documents.

2.4.1.3 Relevance Judgments

Given a collection of documents and a set of search topics, the relevance of each document to each topic needs to be defined to create a test collection. This is not an easy task for two reasons: the complex nature of relevance, and the effort needed to make relevance judgments. Relevance is a subjective measure of relationship between a document and a search topic. It can be affected by many factors including the document's subject, novelty, authority, and availability. Relevance may change with different users and for the same person at different times. In creating IR test collections, the relevance of a document to a search topic is only decided by topical aboutness in a simplified manner.

Given a document collection's size, it is too expensive to make relevance judgments for every document in the collection to every search topic. Therefore, the pooling method selects a subset of documents for each search topic [35]. For a specific search topic, multiple ranked lists are collected from different systems. The top n documents are extracted from each ranked list and merged, with duplicates removed. Relevance judgments for the topic are then limited to this subset of documents. Documents not in this subset are treated as non-relevant. The pooling method greatly reduces the amount of work required to make relevance judgments. The method's usefulness becomes questionable if the number of ranked lists used for pooling is too small or the number of top documents is too small. A study by Zobel and a similar study by Voorhees investigated the effect of pooling method on the quality of the test collection [87, 83]. They concluded the pooling method as used in the TREC test collections is useful.

For practical reasons, there should be a balanced consideration between the number of search topics and the average number of relevance judgments per topic. Buckley and Voorhees conducted a study to investigate the stability of evaluation measures and suggested that 25 search topics are a minimum requirement, and 50 search topics are preferred [9]. Fortunately, the four major IR evaluation workshops have created many test collections that meet this criterion. They are valuable resources for studies such as this dissertation.

2.4.2 Effectiveness Measures

Ideally, all and only the relevant documents are returned by the system. Two basic effectiveness measures are needed to depict the "closeness" of a specific retrieval to the ideal situation. One is the fraction of retrieved relevant documents, which is called *precision*. The other is the fraction of relevant documents that are retrieved, which is called *recall*. That is:

$$precision = \frac{number\ of\ relevant\ documents\ retrieved}{number\ of\ relevant\ documents\ in\ the\ collection} \tag{2.28}$$

$$recall = \frac{number\ of\ relevant\ documents\ retrieved}{total\ number\ of\ documents\ retrieved} \tag{2.29}$$

Precision reflects the "discriminating" power of the system, and recall reflects the "encompassing" power of the system. Since either can give only an incomplete picture of a system, they are used together to calibrate the retrieval effectiveness of an IR system.

Simple recall and precision measures can describe the performance of systems that produce fixed sets of documents, Boolean retrieval systems, which have no relevance ranking. Such simple effectiveness measures are not sufficient to describe performance because both the relevance of a document and its rank are important. Therefore, appropriate measures should take into consideration not only the presence/absence of a relevant document but also its position in the ranked list.

One way to deal with the problem is to compute precision at each retrieved document. In practice, it is neither informative nor necessary to report precision after each retrieved document - a ranked list of 1,000 documents will produce 1,000 such values and it is difficult to either interpret retrieval effectiveness from these 1000 values or compare two retrieval results reported in this way. Therefore, precisions at 11 standard recall levels are used, starting from 0 to 1 with a stepwise increment of 0.1. Plotting precision values at each standard recall level will produce an *11-point precision-recall curve.* For queries that without an actual precision value at a specific standard recall level, the precision is interpolated to the maximum precision for any actual recall level greater than this standard recall level. When a set of search requests are used, average effectiveness can be depicted by average precision over all search requests at each standard recall level. Again, an 11-point precision-recall curve can be used. Curves approaching the top-right corner represent good performance.

We can compute precision at a given document cutoff position in a ranked list, for example, *Precision at 20.* This measure models situations in which users are willing to examine only a fixed set of top-searched documents by the system. A typical example is Web search, for which users usually read only the first one or two retrieved pages of documents. In this case, precision at the 10th retrieved document might be a good measure. In practice, precisions at different document cutoff values give a more complete picture of performance. For example, precisions at 5, 10, 15, 20, 30, 100, 200, 500, 1,000 have been widely used in TREC, CLEF, and NTCIR. Given a set of search topics, average performance can be described by average precision for all search topics at each of these document cutoff values.

Another way of measuring the effectiveness of ranked retrieval system is to use *uninterpolated average precision* for all relevant documents. Starting from the top of a ranked list, whenever a relevant document is encountered, a precision value is computed. When the end of the ranked list is reached, all precision values are summed and divided by the total number of relevant documents for the search topic. Again, when a set of search topics are used, a *mean (uninterpolated) average precision* can be obtained by summing the average precisions of all topics then dividing it by the number of search topics.

Still another measure, *R-precision*, is the precision at the Rth document in a ranked list, where R is the number of relevant document for the topic.

Each of these measures has strengthes and weakness. Precision at standard recall levels makes it possible to compare the performance of the same system with different queries, or to compare the performance of different systems over the same set of queries. The 11-point precision-recall curve provides visualization of the retrieval effectiveness of a system. Intuitively, the area under the curve corresponds to the effectiveness. On the other hand, precisions at standard recall levels often do not correspond to the real precisions of the system because they are interpolated. Mean uninterpolated average precision is a single value measure that makes it simple to quantify differences in retrieval effectiveness. This is also true for other single value measures such as R-precision and precision at a specific retrieved document position (for example, P@10). One weakness of these measures is that they are too general to reflect the detailed differences between two ranked lists.

2.4.3 Interpreting Retrieval Results

Absolute effectiveness measures are difficult to interpret because they rely on the test collections as well as retrieval techniques. It is common to compare the effectiveness of two techniques/systems, assuming one is a comparative condition. There are different ways of interpreting comparison results. Some researchers argue for the use of statistical significance tests. When statistical significance tests are used, first a confidence level is selected, usually either 95% or 99%. Then, the probability of observed difference of retrieval effectiveness between two techniques/systems is computed. If the probability is small enough (less than 0.05 or 0.01), the difference is regarded significant. In other words, the effectiveness of the underlying techniques differs significantly. The two most commonly used statistical significance tests are the paired t-test and the Wilcoxon signed-rank test. Hull discussed the use of statistical significance tests for IR evaluation [31]. Recently reported results from resampling TREC results report 85% confidence for observed differences larger than 10% (relative) at $p < 0.05$ when 25 topics are used, and 90% confidence for 20% relative differences under the same conditions [77].[1] In addition to statistical significance tests, Sparck Jones suggested that a relative difference in evaluation scores of two runs greater than 0.05 absolute is noticeable, and a difference greater than 0.1 absolute is material [33].

In CLIR evaluation, it is common to compare CLIR effectiveness with monolingual baseline, which is obtained by retrieve documents with queries in the same document language (usually translated from the query language by human translators). Monolingual baseline is usually referred to as "upper bound" since it is believed that human translation is the best that automatic translation can get. However, CLIR

[1] Sanderson and Zobel report the best results from a paired t-test, but the Wilcoxon was reported to be nearly as sensitive.

systems sometimes exceed the reported monolingual baseline, which is typically explained by observing that the translation process naturally results in a (possibly beneficial) expansion effect. In an attempt to establish a fair monolingual baseline for this study, we applied a technique that used statistical synonyms derived from statistical translation models to expand monolingual queries (see Section 4.2.1)

2.5 Summary

In this chapter, we reviewed major issues in CLIR and state-of-the-art techniques that have been developed to address them. We note that the majority of research efforts on CLIR have focused on translating queries into document languages. For techniques belonging to this category, structured queries and probabilistic structured queries compute the weight of query terms based on the TF and DF of their translation alternatives. Translation disambiguation techniques selects the correct translations by using term co-occurrence information and/or other sources of evidence. Both types of techniques can significantly improve CLIR effectiveness over baseline conditions, in which neither translation disambiguation nor weighting alternative translations is used.

In addition, we reviewed experimental evaluation of IR and CLIR using test collections. Building a test collection is the key to such evaluation approaches, as it provides a common ground for comparison of different IR systems, models, and techniques. Several contemporary CLIR evaluation workshops have produced proved test collections in more than a dozen languages. Those test collections make new studies possible.

The motivation for this dissertation study derives mainly from three aspects. First, several studies showed using both query translation and document translation could outperform potentially using either alone. We want to further investigate this issue by combining query translation knowledge and document translation knowledge in a more principled way. Second, structured queries and probabilistic structured queries have proved to be an effective in using bilingual dictionaries for CLIR. We want to develop similar, but more effective, techniques that can accommodate knowledge about translation from the document language to the query language. Finally, we want to improve the effectiveness of probabilistic structured queries. We will show all theses issues can be addressed in a general framework for CLIR, which we call cross-language meaning matching. This will be the major topic of the next chapter.

Chapter 3

Matching Meaning for Cross-Language Information Retrieval

In this chapter, we introduce a general framework for CLIR that we call the "meaning matching" model, based on the notion that the goal is to match what the information searcher means with what the document author meant. We use sets of synonyms (synsets), as a computational model of meaning. Given a query term, we compute the probability each document term shares the same meaning by combining two sources of knowledge: (1) statistical term-to-term translations, and (2) synset alignments between the two languages. The meaning matching probability is used for estimating the Term Frequency (TF) and the Document Frequency (DF) of the query term. Finally, the estimated term frequency and document frequency are used for ranking documents. Although the TF and DF computed in this way can be applied to different IR models, we limit our discussions and experiments in the study to the state of the art Okapi weights.

In Section 3.1, we explain the idea of matching meaning for IR in general and for CLIR in specific. The rationale and mathematics of matching meaning between individual terms are introduced in Section 3.2. Specifically, we show how to derive the statistical word-to-synset mapping from statistical word-to-word translations and how to acquire synonymy knowledge. We describe nine variants of the meaning matching model based on the translation knowledge and synonymy knowledge used. Section 3.3 discusses how to use the meaning matching probability between individual terms to estimate the TF and DF of each query term, and how to use those estimates to compute a relevance score for each document. The effectiveness of CLIR based on different variants of meaning matching model is evaluated in Chapter 4 and 5.

3.1 Information Retrieval as Meaning Matching

The goal of IR is to find information relevant to the user's information need, which is expressed as a query. Topical relevance, what we are interested in measuring in this study, can be reflected by the similarity between the meaning of the query and the meaning of the document, or at least some part of the document. One way to estimate the similarity of the meaning between a query and a document is to use the probability that they share the same meaning. In a ranked retrieval system, the probability can then be used to rank documents. Therefore, the task can be simplified as, for a given query Q, to compute $P(Q \leftrightarrow D_i)$, the probability that each document D_i shares the same meaning with Q.

We follow the term independence assumption in IR that the use of each term in a document or in a query is independent of the use of other terms. So, many linguistic features are ignored, and the document and the query are treated as a "bag of words." Under the assumption, matching meaning between a query and a document can be occur through matching meaning between individual query terms and individual document terms. It is well known that in human languages the same terms may have different meanings in different contexts, and different terms may share the same meanings. Furthermore, some meanings are observed more often than others. Therefore, given a query term, there may exist multiple terms in each document that share the same meaning with some probability. To find documents that share the same meaning with a query, we should find not only those that contain terms appearing in the query, but also those that contain terms not appearing in the query but sharing the same meanings with the query terms. More importantly, we should consider the probability that those document terms sharing the same meaning with the query term. We follow tradition, calling terms that share the same meaning *synonyms*.

Meaning matching in most monolingual IR systems is simply based on matching the surface forms or stems of query terms and document terms. With these systems, it will be difficult to find documents that discuss the same topics but use terms different from query terms or do not share the same stems with query terms. Nevertheless, the problem can be mitigated. For example, searchers can choose relevance feedback terms or morphological variants automatically suggested by the system, synonyms from a thesaurus, or try alternative terms based on their subject knowledge [25].

In CLIR, queries and documents are written in different languages, so direct

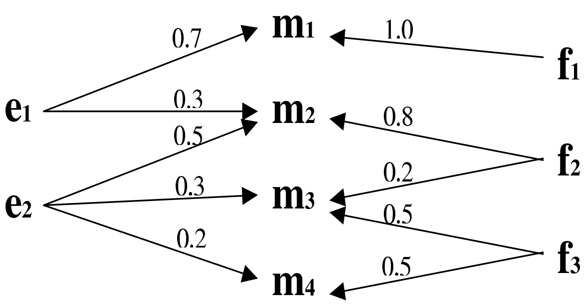


Figure 3.1: Illustrating meaning matching between terms and document terms. e_i denotes query terms, f_j denotes document terms, and m_l denotes meanings.

term matching usually will fail. Matching meaning between query terms and document terms is necessary; given a query term, we want to find its synonyms in each document. From a probabilistic perspective, we need to estimate the probability that each document term sharing the same meaning with the query term.

3.2 Matching Term Meaning across Languages

The core of the cross-language meaning matching model outlined above is: for each term in one language, it specifies the probability that each term in the other language shares the same meaning. To motivate the model's derivation, consider the case in which two English query terms and three French document terms that share subsets of four disjoint meanings, as shown in Figure 3.1. An English query term e_2 has the same meaning as a French document term f_2 if, and only if, e_2 and f_2 express m_2 or e_2 and f_2 express m_3. If we assume the searcher's choice of meaning for e_2 is independent of the author's choice of meaning for f_2, we can compute probability that e_2 and f_2 share the same meaning. Generalizing to any pair of words e and f:

$$
\begin{aligned}
p(e \leftrightarrow f) &= \sum_{m_i} p(m_i|e, f) \\
&\cong \sum_{m_i} p(m_i|e) \times p(m_i|f)
\end{aligned}
\tag{3.1}
$$

where:

- $p(e \leftrightarrow f)$: the probability that term e and term f have the same meaning.
- $p(m_i|e)$: the probability that term e has the meaning m_i
- $p(m_i|f)$: the probability that term f has the meaning m_i

Based on this formula, we can compute the meaning matching probability distribution for each pair of query-document terms in Figure 3.1 as follows:

$$p(e_1 \leftrightarrow f_1) = p(m_1|e_1) \times p(m_1|f_1) = 0.7 \times 1.0 = 0.7$$

$$p(e_1 \leftrightarrow f_2) = p(m_2|e_1) \times p(m_2|f_2) = 0.3 \times 0.8 = 0.24$$

$$p(e_1 \leftrightarrow f_3) = 0$$

$$p(e_2 \leftrightarrow f_1) = 0$$

$$p(e_2 \leftrightarrow f_2) = p(m_2|e_2) \times p(m_2|f_2) + p(m_3|e_2) \times p(m_3|f_2) = 0.5 \times 0.8 + 0.3 \times 0.2 = 0.46$$

$$p(e_2 \leftrightarrow f_3) = p(m_3|e_2) \times p(m_3|f_3) + p(m_4|e_2) \times p(m_4|f_3) = 0.3 \times 0.5 + 0.2 \times 0.5 = 0.25$$

Meaning matching probabilities computed this way are not normalized. That is, for a given term in one language, the summation of the probability of its translations in the other language may not equal to 1. As a result, some terms may receive more emphasis than others after translation is performed. This could be a problem because we expect meanings should be independent of languages used to convey the meanings. In other words, the meaning of a term in one language should completely be able to be expressed in another language. Therefore, we need to normalize the meaning matching probability, dividing the raw meaning matching probability of each translation with the summation of all probabilities, so that:

$$\sum_{f_i} p(e \leftrightarrow f_i) = 1 \tag{3.2}$$

or

$$\sum_{e_i} p(e_i \leftrightarrow f) = 1 \tag{3.3}$$

As the above equations show, probability normalization can be done either on the query language side or on the document language side. If we normalize meaning matching probability on the document language side, for a given query term, the summation of the meaning matching probability of each of its synonyms in the document language will be 1. On the other hand, if probability normalization is performed on the query language side, it means for a given document term, the summation of the meaning matching probability of each of its synonyms in the query language will be 1. In deciding which side the probabilities will be normalized, the way that the probabilities should be considered. For unidirectional translation knowledge which is asymmetric in nature, it is desirable to perform normalization in the target language. For bidirectional translation knowledge (i.e., using both query translation knowledge and document translation knowledge), probability normalization can be done in either way. In this study, we normalized translation probabilities on the document language side when bidirectional translation knowldge was used.

Now the problem of meaning matching becomes how to develop a computational model of meaning (m_i) and how to compute the probability of a term having a meaning (i.e., $p(m_i|e)$ and $p(m_i|f)$).

3.2.1 Using Synsets to Represent Meanings

A simple way to represent meaning uses sets of synonymous terms. We call each set a synset. In this case, some source of synonymy knowledge must be available, and each synset is assumed to encode a different meaning.

We can use existing synonymy knowledge resources such as synonym dictionaries, thesauri, and WordNet [55]. In recent years, WordNet has become an important resource for natural language processing. In WordNet, English nouns, verbs, adjectives and adverbs are organized into synsets, each representing one underlying lexical concept. Therefore, WordNet can be used to group individual English words into synsets. Synonym dictionaries provide a set of synonymous terms for a given term. We can group a term with its synonyms to form a synset. Thesauri are much like synonym dictionaries, although they may contain more information such as broader terms and narrower terms in addition to synonymy knowledge. They can be used in the same way as synonym dictionaries for the task.

Synonyms can also be produced statistically from corpus. As described in

Section 2.3.2, translation models in both directions can be derived by applying statistical approaches on a sentence-aligned corpus. For a word e in Language E, we develop a model that specifies the probability that it translates into each word $f_i(i = 1, 2, \ldots, n)$ in Language F. Likewise, for a word f in Language F, we develop a model that specifies the probability that it translates to each word $e_j(j = 1, 2, \ldots, m)$ in Language E. We then combine these two translation models to derive a statistical synonym model. Specifically, we assume the probability of word f_j, being a synonym of word f, can be approximated by multiplying the probability of word f translating to some word e_i and the probability of word e_i translating to word f_j. Mathematically,

$$p(f_j|f) \cong \sum_{i=1}^{n} p(e_i|f) \times p(f_j|e_i) \tag{3.4}$$

where $p(f_j|f)$ refers to the probability of f_j being a synonym of f. Using this formula, a synonym set $f_j(j = 1, 2, \ldots, n)$ of word f can be created and ranked. Since the resulting list contains many terms with low probabilities that are difficult to accurately estimate, it is reasonable to select only candidate synonyms with relatively high probabilities.

3.2.2 From Statistical Translation to Word-to-Synset Mapping

With the synonymy knowledge described above, we can develop a probabilistic model that maps words in Language E to synsets in Language F, or vice versa. Considering the mapping of words in Language E to synsets in F, our approach combines statistical word-to-word translation model from Language E to Language F with synonymy knowledge in Language F. Again, a statistical word-to-word translation model $p(f_i|e)$ $(i = 1, 2, \ldots, n)$ specifies, for a given word e in Language E, a set of words f_i $(i = 1, 2, \ldots, n)$ in Language F and the probability that e translates to each of f_i. With the translation model, we can group the n translation alternatives of e into sets of synonymous translations using synonymy knowledge in Language F. The process of conflating multiple translation alternatives of a term into synsets and estimating the probability that the term maps to each synset is called *aggregation* in this dissertation. To aggregate a set of translation alternatives into synsets, we assume each term in Language E could have multiple meanings, and hence its translations in Language F should be grouped so that each resulting synset denotes one of its meanings. In the special case that the term has only one meaning, all its translations should be grouped into one synset.

One term possibly may appear in multiple synsets. In this case, we decide to which synset or synsets to assign when performing aggregation. Figure 3.2 shows two methods, with an example. In Figure 3.2(a), each translation of word e is assigned to each of its synsets with uniform probability. For example, since translation f_1 appears in two synsets S_1 and S_2 and the translation probability from e to f_1 is 0.2, the mapping probability from e to each of these two synsets will be $0.2 \div 2 = 0.1$.

In Figure 3.2(b), each translation is limited to be assigned to only one synset. If a translation appears in multiple synsets, we decide to which synset it will go eventually. We apply an aggregation heuristic based on the maximum cumulative probability of each synset. Specifically, all the possible synsets first are found as

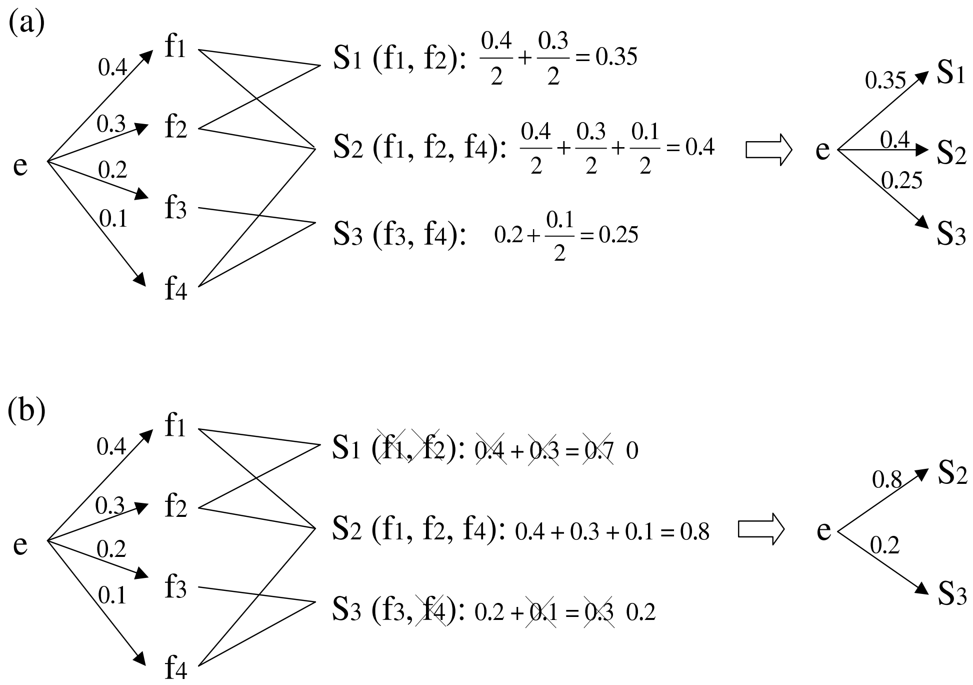

(a)
e
0.4
0.3
0.2
0.1
f1
f2
f3
f4
S1 (f1, f2): $\frac{0.4}{2}+\frac{0.3}{2}=0.35$
S2 (f1, f2, f4): $\frac{0.4}{2}+\frac{0.3}{2}+\frac{0.1}{2}=0.4$
S3 (f3, f4): $0.2+\frac{0.1}{2}=0.25$
e
0.35
0.4
0.25
S1
S2
S3
(b)
e
0.4
0.3
0.2
0.1
f1
f2
f3
f4
S1 (~~f1~~, ~~f2~~): ~~0.4~~ + ~~0.3~~ = ~~0.7~~ 0
S2 (f1, f2, f4): 0.4 + 0.3 + 0.1 = 0.8
S3 (f3, ~~f4~~): 0.2 + ~~0.1~~ = ~~0.3~~ 0.2
e
0.8
0.2
S2
S3

Figure 3.2: Two methods of conflating multiple translations into synsets. (a) conservative method. (b) greedy method. f_i $(i = 1, 2, 3, 4)$: translations of term e, S_j $(j = 1, 2, 3)$: synsets.

we did with Method (a). Then the synset with the largest cumulative probability (i.e., the summation of the probability that e translates to each of the term in the synset) is selected. Translations in this selected synset are excluded from remaining synsets, consequently recomputing the mapping probabilities of remaining synsets. Repeating these two steps will eventually produce the final word-to-synset mappings. The following algorithm describes this method:

1. Search the synonym space in Language B for synsets pool $S_j (j = 1, 2, \ldots, m)$ such that each S_j contains at least one of the translations of e;

2. Compute the probability that e maps to S_j: $p(S_j|e) = \sum_{f_k \in S_j} p(f_k|e)$, and rank all S_j in the decreasing order of the probability computed in this way;

3. Select the top ranked synset (and its probability), exclude it from the synsets pool, and exclude translations contained in this synset from all the remaining synsets in the pool;

4. Repeat Step 2 and 3 until all synsets remaining in the pool are empty or no synset remains in the pool. The selected synsets together with their probability form the word-to-synset mapping model for term e.

Which aggregation method is better? Method (a) is a *conservative* approach because it attempts to include every possible synset in the final result. As a result, the final number of synsets with Method (a) could in some cases be larger than the number of the translations. On the other hand, Method (b) is a *greedy* approach as it attempts to include as many translations in as few synsets as possible. With this method, the number of resulting synsets will never exceed the number of translation alternatives. Also, the probability mass distribution among synsets in Method (a) is more balanced, whereas the probability mass distribution among synsets in Method (b) is more skewed. Statistical translation models trained with parallel corpora contain many translation alternatives, and some low-probability translations may be incorrect. However, if a low-probability translation co-occurs with a high-probability translation, it may be a correct translation; if it never co-occurs with any high-probability translation, it may be wrong. With the greedy method, correct translations with low probability are more likely to be grouped into synsets with high-aggregated probability. Therefore, we adopted the greedy method.

3.2.3 Variants of Meaning Matching

We have discussed how to acquire synonymy knowledge and how to use synsets to represent meanings and to convert word-to-word translation into word-to-synset mapping. We will develop the term-to-term meaning matching model based on which translation knowledge and/or synonymy knowledge is used.

Table 3.1 shows nine variants of probabilistic meaning matching in which we are particularly interested. The differences among these models can be viewed from three perspectives: (1) whether the translation knowledge used is bidirectional or unidirectional, and if unidirectional, whether it is from query language to document language or the reverse, (2) whether synonymy knowledge is used in both languages or only one, and if in one language, whether it is in the query or document language,

	Icon	Query translation knowledge	Document translation knowledge	Query language synsets	Document language synsets	Pre-aligned synsets
FAMM	$(Q) \Leftrightarrow (D)$	✓	✓	✓	✓	✓
DAMM	$(Q) \rightleftharpoons (D)$	✓	✓	✓	✓	
PAMM-q	$(Q) \rightleftharpoons D$	✓	✓	✓		
PAMM-d	$Q \rightleftharpoons (D)$	✓	✓		✓	
IMM	$Q \rightleftharpoons D$	✓	✓			
PSQ	$Q \rightarrow D$	✓				
PDT	$Q \leftarrow D$		✓			
APSQ	$Q \rightarrow (D)$	✓			✓	
APDT	$(Q) \leftarrow D$		✓	✓		

Table 3.1: Variants of term-to-term meaning matching. FAMM: Full Aggregated Meaning Matching; DAMM: Disconnected Aggregated Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; IMM: Individual Meaning Matching; *PSQ*: Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APSQ: Aggregated Probabilistic Structured Queries; APDT: Aggregated Probabilistic Document Translation. Meanings of icons: arrows represent translation directions; parentheses represent aggregation.

and (3) whether synsets are aligned across languages.

3.2.3.1 Full Aggregated Meaning Matching

Ideally, we want words in the two languages mapped to the same meaning space, or to the same synset space as we are representing meanings with synsets. How do we map words in different languages into the same synsets? Given each synset consists of synonyms, it should contain words in both languages. Each synset should contain two sub-synsets, each of which is in one of the two languages; they are a pair of aligned synsets.

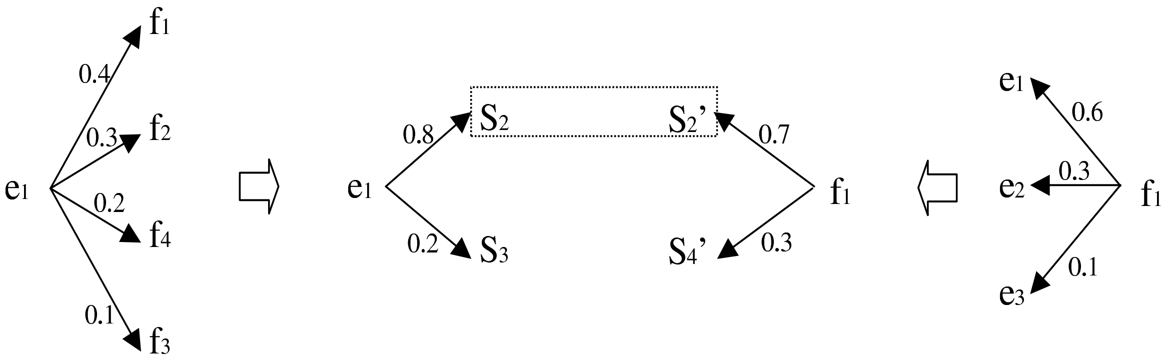


Figure 3.3: Full aggregated meaning matching.

Figure 3.3 illustrates the idea of mapping terms in one language to synsets in the other language, and linking the synsets with existing knowledge of synset alignment. We call this method “full aggregated meaning matching” (FAMM). In the example shown in the figure, four translations of word e_1 are conflated into two synsets S_2 and S_3 using the greedy method. Similarly, three translations of word f_1 are conflated into two synsets S'_2 and S'_4. For the simplicity of discussion, we assume e_1 is in the query language and f_1 is in the document language (same below). The only aligned pair among four possible alignments is (S_2, S_2'), meaning only this pair contributes to the meaning matching between e_1 and f_1. Specifically, the probability that e_1 and f_1 share the same meaning is:

$$p(e_1 \leftrightarrow f_1) = p(s_2|e_1) \times p(s_2\prime|f_1) = 0.8 \times 0.7 = 0.56$$

Generalizing to any pair of words e and f, the full aggregated meaning matching model can be expressed as:

$$p(e \leftrightarrow f) = \sum_{(s_{e_i}, s_{f_i}) \in s_i} p(s_{e_i}|e) \times p(s_{f_i}|f) \tag{3.5}$$

where

- s_{e_i}: the ith synset of the translations of word e,
- s_{f_i}: the ith synset of the translations of word f,
- s_i: the set of all aligned synsets s_{e_i} and s_{f_i}.

One unique feature of FAMM is that it allows multiple alignments of synsets between any pair of words. A polysemous word in one langauge and a polysemous word in another language may share more than one meaning. For example, the English word “drug” and the Chinese word “Yao” can both mean “medicine” and “illegal drug”.

However, there are at least two practical limitations to FAMM. First, resources containing synsets aligned across languages are rare. EuroWordNet is the only resource we had with this property. For languages not covered by EuroWorldNet, FAMM would require the development of aligned synsets between two languages. Second, term coverage by the aligned synsets may be a serious problem. As we will show later in Chapter 4, many important terms are not covered by EuroWordNet, which significantly degraded the effectiveness of CLIR based on FAMM.

3.2.3.2 Disconnected Aggregated Meaning Matching

Full aggregated meaning matching requires synset alignments, which are not readily available for many language pairs. Disconnected aggregated meaning matching (DAMM) is developed to deal with the situation in which translations in both directions and synsets in both languages are used, but synset alignment knowledge is not available. "Disconnected" should be interpreted only as "there is no fixed one-to-one synset alignment available." In order to use the available translation knowledge and synonymy knowledge, we assume a query word and a document word have the same meaning only if each word appears in one of the other word's translation synsets.

Figure 3.4 shows how DAMM is realized. We start with two words, e_1 and f_1, and we want to develop a model that specifies the probability that they share the same meaning. The greedy method groups synonymous translations of each word into synsets. Next, we traverse the resulting synsets of word e_1 to find a synset that contains word f_1. If we succeed, we reference word e_1 to the translation synsets of word f_1. If we succeed again then the meaning matching between e_1 and f_1 succeeds. If we fail at either of these two steps, we assume e_1 and f_1 do not share the same meaning. In the example in Figure 3.4, we found synset s_2 of e_1 contains f_1, while

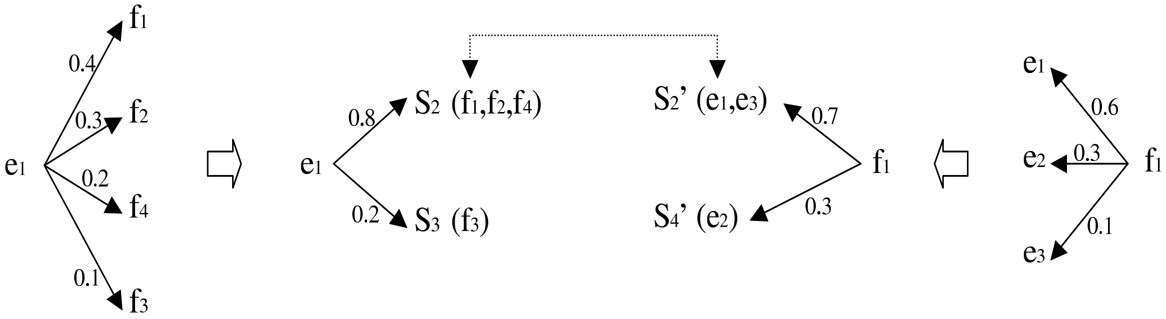


Figure 3.4: Disconnected aggregated meaning matching.

synset $s_{2'}$ of f_1 contains e_1, so meaning matching between these two words succeeds. Specifically, the meaning matching probability between e_1 and f_1 is:

$p(e_1 \leftrightarrow f_1) = p(s_2|e_1) \times p(s_{2'}|f_1) = 0.8 \times 0.7 = 0.56$

Generalizing to any pair of words e and f, the disconnected aggregated meaning matching model can be expressed as:

$$p(e \leftrightarrow f) = p(s_{e_i}|e) \times p(s_{f_j}|f) \quad (f \in s_{e_i}, e \in s_{f_j}) \tag{3.6}$$

Although FAMM and DAMM look quite similar (see Equation 3.5 and Equation 3.6), they are different. Due to the lack of synset alignment information, an extra assumption must be made for DAMM. When using the greedy synonym conflation method, no more than one pair of aligned synsets can be found between any pair of words in DAMM. This is why Equation 3.6 does not have summation.

3.2.3.3 Partial Aggregated Meaning Matching

When bidirectional translation knowledge and synonymy knowledge in only one language are used, we call the variant "Partial Aggregated Meaning Matching"

(PAMM). Aggregation is partial because it is done in only one of the two languages. In this case, we assume each individual translation encodes a different meaning when aggregation is not performed on the translation side. Under this assumption, meaning matching can be realized by linking translation synsets on one side to translations on the other side.

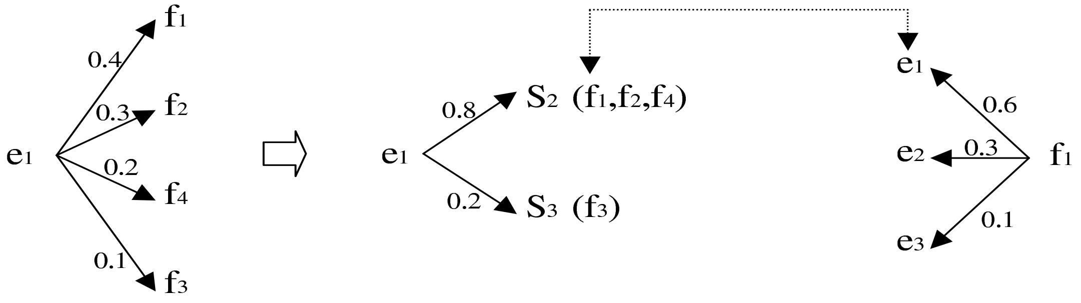


Figure 3.5: Partial aggregated meaning matching: aggregated in document language.

Figure 3.5 illustrates the situation in which only synonymy knowledge in the document language is available. In the example, query word e_1 is translated, and its translations are conflated into synsets with the greedy method based on synonymy knowledge in document language. Meanwhile, document word f_1 is also translated. We then check if f_1 appears in any translation synset of e_1 and e_1 is a translation of f_1. In the example, f_1 appears in synset s_2 of e_1, and e_1 is a translation of f_1. Therefore, meaning matching between e_1 and f_1 succeeds. Specifically, the meaning matching probability between these two words is:

$p(e_1 \leftrightarrow f_1) = p(s_2|e_1) \times p(e_1|f_1) = 0.8 \times 0.6 = 0.48$

Generalizing to any pair of word e and f, the model of PAMM in document language (PAMM-d) can be expressed as:

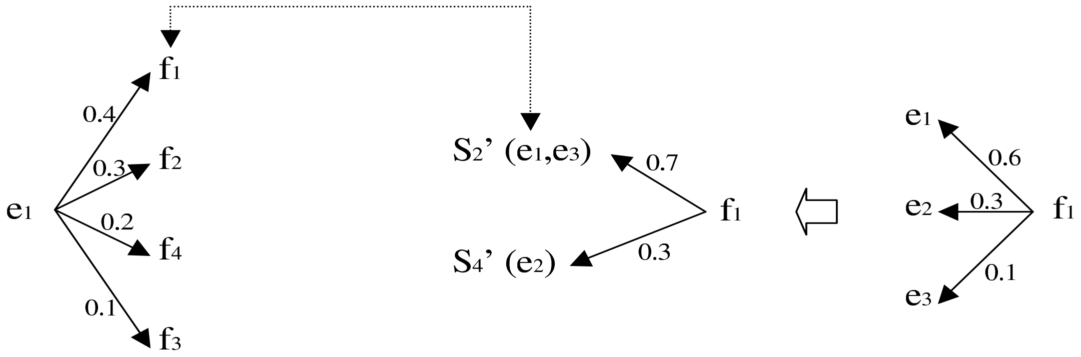


Figure 3.6: Partial aggregated meaning matching: aggregated in query language.

$$p(e \leftrightarrow f) = p(s_{e_i}|e) \times p(e|f), \ f \in s_{e_i} \tag{3.7}$$

The method shown in the example is called PAMM-d, in which the conflation of synonymous translations is done on the document side.

When synonymy knowledge is available in the query language instead of the document language, we can make the same assumption that each word in the document language represents a different meaning. As shown in Figure 3.6, translations of document word f_1 are conflated into synsets using the greedy method. Notice that query word e_1 is contained in translation synset $s_{2\prime}$ of f_1 and word f_1 is a translation of e_1. Therefore, meaning matching between word e_1 and word f_1 succeeds. The meaning matching probability is:

$p(e_1 \leftrightarrow f_1) = p(f_1|e_1) \times p(s_{2\prime}|f_1) = 0.4 \times 0.7 = 0.28$

Generalizing to any pair of word e and f, the model of partial meaning matching aggregated in query language can be expressed as:

$$p(e \leftrightarrow f) = p(f|e) \times p(s_{f_i}|f), \ e \in s_{f_i} \tag{3.8}$$

The situation is called Partial Aggregated Meaning Matching in the query language (PAMM-q), in which the conflation of synonymous translations is done on the query side.

Notice with statistical translation models learned from parallel corpus, we can always obtain statistical synonymy knowledge in both languages. In this case, the premise of PAMM is artificial as we can always use DAMM if it helps. However, if other types of synonymy knowledge such as WordNet are used, synonymy knowledge may be available in only one language.

3.2.3.4 Individual Meaning Matching

Accurate computational models of synonymy knowledge may not be available, especially for language pairs that are not well-studied. When synonymy knowledge in neither language, is available or is not used, we adopt the same assumption that each word encodes a different meaning. In this case, two words share the same meaning only if they are translations of each other. The probability that two words share the same meaning is dependent on the probability that e translates to f and the probability that f translates to e, i.e.,

$$p(e \leftrightarrow f) = p(e|f) \times p(f|e) \tag{3.9}$$

Figure 3.7 illustrates meaning matching between word e_1 and word f_1 in this situation. The meaning matching probability can be simply computed as:

$p(e_1 \leftrightarrow f_1) = p(f_1|e_1) \times p(e_1|f_1) = 0.4 \times 0.6 = 0.24$

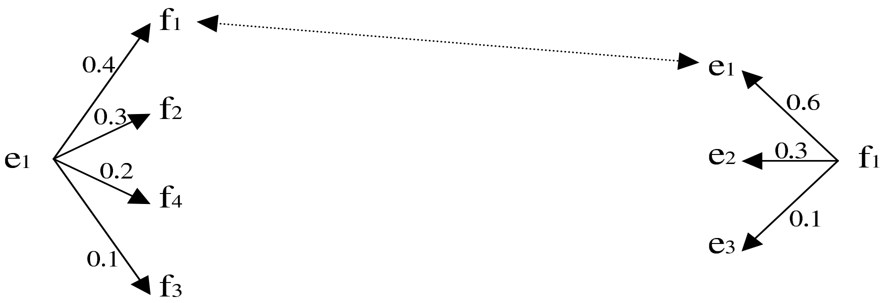


Figure 3.7: Individual meaning matching.

This model is called *Individual* Meaning Matching (IMM) since only individual words are involved, due to the lack of synonymy knowledge.

3.2.3.5 Probabilistic Structured Queries

We have focused on the use of synonymy knowledge, assuming statistical translation knowledge is available in both directions. We now look at the situations in which translation knowledge is available only in one direction. In this case, if synonymy knowledge is not available, meaning matching becomes a unidirectional translation model. If only translation knowledge from query language to document language is available, the probability of a query word and a document word sharing the same meaning is the probability of the query word translating to the document word:

$$p(e \leftrightarrow f) = p(f|e) \tag{3.10}$$

For the same pair of words e_1 and f_1 in the illustrative example, the meaning matching probability is:

$p(e_1 \leftrightarrow f_1) = p(f_1|e_1) = 0.4$

This is the Probabilistic Structured Query (PSQ) method that was first introduced by Darwish [15]. Notice it is an extreme case of individual meaning matching in which we assume the probability that a document word translates to a query word is 1.

3.2.3.6 Probabilistic Document Translation

If only translation knowledge from document language to query language is available, meaning matching becomes translation from document language to query language. In this case, we assume the probability that two words share the same meaning could be approximated by the probability that the document word translates to the query word:

$$p(e \leftrightarrow f) = p(e|f) \tag{3.11}$$

The meaning matching probability between the same pair of words in the above example is:

$p(e_1 \leftrightarrow f_1) = p(e_1|f_1) = 0.6$

We call this model Probabilistic Document Translation (PDT) as it is analog to PSQ but applied on the document side. It is another extreme case of individual meaning matching in which the probability of a query word translating to a document word is 1.

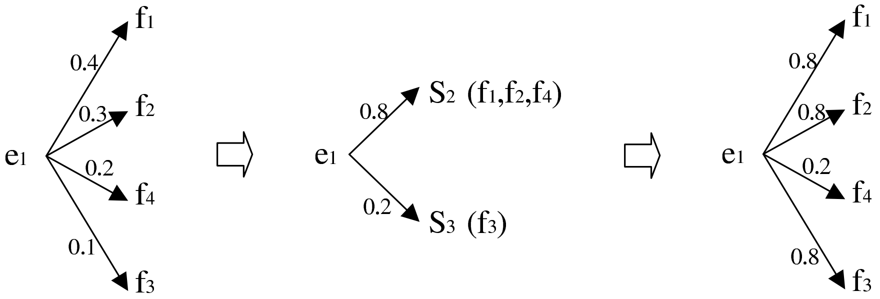


Figure 3.8: Aggregated probabilistic structured queries.

3.2.3.7 Aggregated Probabilistic Structured Queries

In another situation of meaning matching, a unidirectional translation model combines with synonymy knowledge on the translation side. One variant involves query translation coupling with aggregation in document language. The probability a query word and a document word share the same meaning is computed as the probability that the query word maps to the synset that the document word belongs to. We call the meaning matching Aggregated Probabilistic Structured Queries (APSQ) because it translates queries and conflates translations into synsets based on synonymy knowledge in document language. Mathematically, meaning matching in this case is described as:

$$p(e \leftrightarrow f) = p(s_i|e) \quad (f \in s_i)$$

In the example in Figure 3.8, the meaning matching probability between e_1 and f_1 is:

$$p(e_1 \leftrightarrow f_1) = p(s_2|e_1) = 0.8$$

APSQ is an extreme case of PAMM-d in which the probability of a document word translating to a query word is treated as 1.

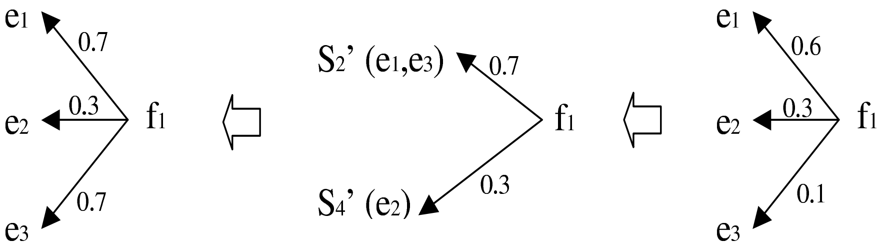


Figure 3.9: Aggregated probabilistic document translation.

3.2.3.8 Aggregated Probabilistic Document Translation

The last variant of meaning matching uses document translation knowledge and synonymy knowledge in the query language. In this case, the probability a query word and a document word share the same meaning is estimated as the probability that the document word maps to the synset to the query word belongs. We call this model Aggregated Probabilistic Document Translation (APDT) because it translates documents and aggregate translations into synsets with synonymy knowledge in the query language. Mathematically, meaning matching in this case can be expressed as:

$p(e \leftrightarrow f) = p(s_j|f),\ e \in s_j$

In Figure 3.9, the meaning matching probability between e_1 and f_1 is:

$p(e_1 \leftrightarrow f_1) = p(s_{2'}|f_1) = 0.7$

In APDT, the probability that a query word translates to a document word is treated as 1.

The premise of meaning matching with unidirectional translation knowledge is artificial when the translation knowledge is obtained from parallel corpus, as we can acquire translation knowledge in both directions and statistical synonymy knowledge in both languages. We discuss it here for two reasons. First, other situations exist in which a probabilistic bilingual dictionary is only available in one direction. Second, we want to show the best existing query translation technique (in this case PSQ) is a special case of our meaning matching model. It can serve as a reference for comparison with other variants of meaning matching model in which more translation knowledge and/or synonymy knowledge are used. Our hypothesis is that by using more knowledge, the meaning matching model can achieve better CLIR effectiveness. In Chapter 4 and 5, we will present comparative experiment results regarding the effectiveness of these variants.

3.3 Ranking Documents by Term Weights

In Section 3.2, we provided the derivations of meaning matching between individual query terms and document terms under different situations. Regardless of the situation, we use $p(e \leftrightarrow f)$ to denote the probability query term e and document term f share the same meaning. In this section, we look at how the meaning matching probability can be used to estimate a weight for a query term in each document. After we estimate a weight for each query term in each document, then we can compute a Retrieval Status Value (RSV) for each document, and use it for ranking documents.

In a TF/DF based retrieval system, the weight of a query term can be computed by combining its TF and DF. In monolingual IR, since queries and documents are written in the same language, the TF and DF of each query term can be obtained by counting the occurrences of the term in each document and the total number of documents that contain the term. In CLIR, however, we have to *estimate* the TF and DF of each query term in each document since the query term usually does not appear in any document.

3.3.1 Estimating TF and DF

According to the meaning matching model, for a given word e in Language E, we find a set of terms f_i $(i = 1, 2, \ldots, n)$ in Language F, each of which shares the same meaning with term e with some probability $p(e \leftrightarrow f_i)$ $(i = 1, 2, \ldots, n)$ respectively. We continue to call each f_i a translation of e. If we take Language E as the query language and Language F as the document language, then for any query term a set of translations exists in the document language. Whenever we see a translation f_i appearing one time in document d_k, we assume we have seen term e for p_i times. Suppose the total occurrence of term f_i in document d_k is $TF(f_i, d_k)$, the total "occurrence" of term e as estimated from the occurrences of term f_i will be $p(e \leftrightarrow f_i) \times TF(f_i, d_k)$. Considering all the translations of term e in document d_k, we have our estimate of $TF(e, d_k)$ as the term frequency of query term e in document d_k:

$$TF(e, d_k) = \sum_{f_i} p(e \leftrightarrow f_i) \times TF(f_i, d_k) \tag{3.12}$$

For the DF of query term e, as long as document d_k contains any of its translations f_i, we assume the document "contains" e too. When meaning matching probability is *not* considered, we can use the total number of documents that contain at least one synonym of term e to estimate its DF, which is exactly the way that DF is estimated in the structured query method (Equation 2.23). However, intuitively translation probability should also be used in the estimation of document frequency. If all the translations of the query term appear in the document, its DF should add 1; if only some of its translations appear in the document, its DF should add less than 1. In the latter case, it is reasonable to use the summation of the probability of its translations appearing in the document. Considering all documents that contain at least one of its translations, the DF of query term e in the collection can be estimated as:

$$DF(e) = \sum_{f_i} p(e \leftrightarrow f_i) \times DF(f_i) \tag{3.13}$$

This approach to estimating DF is very similar to that used in PSQ (Equation 2.26), except that the probability used here is the meaning matching probability instead of a unidirectional translation probability.

3.3.2 Combining Query Term Weights

After the TF and DF of a query term are estimated, we can compute its weight by combining the TF and DF with other factor such as normalized document length. Theoretically, the TF and DF estimated in this way can be used in any TF/DF-based IR model, such as the vector space model. The experiments reported later in the dissertation were conducted with the state of the art Okapi BM25 formula [74].

We used $k_1 = 1.2$, $b = 0.75$, and $k_3 = 7$ as has been commonly used. Specifically, the weighting function we used is:

$$RSV(Q, d_k) = \sum_{e \in Q} [\log \frac{(N - df(e) + 0.5)}{(df(e) + 0.5)}][\frac{(2.2 * tf(e, d_k))}{(0.3 + 0.9 * \frac{dl(d_k)}{avdl} + tf)} \frac{(8 * qtf(e))}{(7 + qtf(e))}] \tag{3.14}$$

where:

- $RSV(Q, d_k)$ is the retrieval status value of document d_k concerning query Q; it is the weight of the document.
- N is the total number of documents in the collection.
- $df(e)$ is the document frequency of e, as estimated using Equation 3.13.
- $tf(e, d_k)$ is the term frequency of e in d_k, as estimated using Equation 3.12.
- $dl(d_k)$ is the length of document d_k.
- $avdl$ is the average document length in the collection.
- $qtf(e)$ is the term frequency of e in Q.

The equation first computes the weight of each query term and then sums the weight of all query terms. The result is the retrieval status value of the document regarding the query. Given a query, we compute the value for every document in the collection. Documents then can be ranked by their retrieval status values, achieving ranked retrieval.

3.3.3 Pruning Translations in Meaning Matching

The number of translation alternatives gained from training statistical translation models could become enormous, as increasingly unlikely translations are included. For this reason, it is important understand the sensibility of CLIR effectiveness to the degree of translation fanout. In some early studies on dictionary-based CLIR, the issue was investigated by changing the number of translation alternatives. For translations obtained from statistical training, this may not be a good idea because it ignores translation probability in choosing translations. As a result, for example, the top two translations of some terms may be likely translations, whereas the top two translations of other terms may include an unlikely translation. Therefore, the meaning matching probability should be used in pruning translations. In one way, an Individual Probability Threshold (IPT) is set with any translations whose probability is below the IPT excluded.

Another way to study the relationship between the degree of translation fanout and CLIR effectiveness is the Cumulative Probability Threshold (CPT) [15]. CPT method first ranks all translations in decreasing order of their probabilities, then selects translations top-down until the cumulative probability of the selected translations first reaches or exceeds a pre-set threshold. This way, a CPT of 0 ensures the use of top 1 translation, a CPT of 1 ensures the use of all translation alternatives, and a CPT between 0 and 1 ensures the use of some translation alternatives with relatively high probability. By setting different CPT values between 0 and 1, we will see how CLIR effectiveness changes as the richness of translations changes.

IPT and CPT have similar effects on the selection of translation alternatives. Both use probabilities instead of counts in deciding which translations to use. For this reason, they may lead to the use of different numbers of translation alternatives for different terms. In addition, if the probability of included translations is re-normalized, neither method will alter the computation of document length in term weighting. Since the study by Darwish and Oard showed CPT was useful for studying the sensitivity of CLIR to translation fanout, we adopted it in our study reported in this dissertation [15].

3.4 Summary

In this chapter, we introduced the idea of relating information retrieval to matching the meaning of query and document terms. Based on this idea, we developed the meaning matching model that specifies the probability that a query term and a document term share the same meaning. In this model, meanings are represented by synsets, and meaning matching is accomplished by mapping query terms and document terms to aligned synsets. When the knowledge of synset alignment is unavailable, we made assumptions that allowed us to use the available translation knowledge and/or synonymy knowledge. We described how the meaning matching probability can be used for ranking documents in CLIR. Issues related to the implementation of the model and its robustness to the degree of translation fanout were discussed.

In Chapters 4 and 5, we present results of experiments in which the meaning matching model was used. The experiments involve three test collections for two language pairs. Specifically, we look at the effectiveness of CLIR based on the nine variants of the meaning matching model, and show that using bidirectional translation knowledge and synonymy knowledge led to significant improvement of CLIR effectiveness.

Chapter 4

English-French CLIR Experiments

To study the effectiveness of the meaning matching model developed in the previous chapter, two sets of experiments were conducted. The first set of experiments concerns the retrieval of French documents for information requests expressed in English, and the second set of experiments focuses on finding Chinese documents in response to information requests written in English. In this chapter, we present the English-French CLIR experiments, and the English-Chinese CLIR experiments in the next chapter.

In Section 4.1, we describe our experiment design, which includes selecting the test collection and IR system, query/document processing, training statistical translation models, deriving statistical synonyms, and selecting an effectiveness measure. We then present the experiment results and detailed analysis in Section 4.2. We compare meaning matching effectiveness with a monolingual baseline, investigate different variants of meaning matching, and compare their effectiveness when different statistical MT training setups were used. Section 4.3 summarizes our findings.

4.1 Experiment Design

4.1.1 Test Collection and IR System

We combined the French test collections created by the Cross-Language Evaluation Forum (CLEF) in 2001, 2002 and 2003 into a single collection. The collection contains 87,191 French news articles from Le Monde and SDA in 1994, 151 topics in English (and, for comparison, in French),[1] and binary relevance judgments created using a pooled assessment methodology. We stripped accents from the document collection and removed French terms contained on the stopword list provided with the open source Snowball stemmer.[2] We then created two document indexes: one based on unstemmed French terms and the other with stemmed French terms using the Snowball French stemmer (with accents removed).[3]

Each topic contains title, description, and narrative fields. We automatically created two types of queries for each topic by using words from the title field alone (*T* queries, representing a brief Web-like query) or words from title and description fields (*TD* queries, representing a brief sentence or two a searcher might say when approaching a librarian for assistance). For French queries, we performed accent removal, stopword removal, and stemming using the same tools we used for processing

[1] The 9 of the 160 CLEF 2001, 2002 and 2003 topics for which no relevant French documents are known were removed from the collection.

[2] http://snowball.tartarus.org/

[3] This is post-accent-removal stemming. A better method is pre-accent-removal stemming. We compared every rule contained in Snowball French stemmer and found only one case that made a minor difference between these two stemming methods.

the document collection. For English queries, we performed pre-translation stopword removal using the English stopword list provided with the Inquery retrieval system obtained from the University of Massachusetts.

The experiments were run with the Perl Search Engine (PSE), which uses Okapi BM25 weights (see Equation 3.14). We modified it to implement other variants of the meaning matching model.

4.1.2 Training Statistical Translation Models

We used the freely available GIZA++ toolkit [64][4] to train translation models with the Europarl parallel corpus [38]. Europarl contains 677,913 automatically aligned sentence pairs in English and French from the European Parliament. We stripped accents from every character and filtered out implausible sentence alignments by eliminating sentence pairs with a token ratio either smaller than 0.2 or larger than 5, resulting in 672,247 sentence pairs for actual use. Word alignment models implemented by GIZA++ are sensitive to the translation direction, so we ran GIZA++ twice, once with English as the source language and once with French as the source language. In both cases, we started with 10 IBM Model 1 iterations, followed by 5 HMM iterations, and ending with 5 IBM Model 4 iterations. The "alignment templates" technique was not used, so all alignment pairs include a single word in each language. GIZA++ produces a representation of the sparse translation matrix using a three-column table that specifies, for each source-target word pair, the normalized translation probability of the target language word given the source language word.

[4] http://www-i6.informatik.rwth-aachen.de/Colleagues/och/software/GIZA++.html

With the two translation models from GIZA++, we identified statistical synonyms and derived all variants of the meaning matching model described in Chapter 3.

4.1.3 Identifying Synonyms

Synonyms used in the experiments were acquired from three sources: WordNet (for English), EuroWordNet (for aligned synsets between English and French), and the parallel corpus (for both English and French). We downloaded the freely distributed WordNet 2.0 from its project Web site and used a Perl module to extract synsets from it.[5]. In EuroWordNet, each synset is identified by a unique synset id, and under the same id, terms in several European languages are listed separately. Therefore, aligned synsets were extracted easily.

To create a statistical synonym dictionary, we started with the two statistical translation models derived from statistical MT training with GIZA++ as described above. We then used Equation 3.4 to find synonyms and their probability for all terms in both languages. Synonyms with a normalized probability lower than 0.1 were excluded from the lists.

4.2 Results

In this section, we report our experiment results. We present the results in three parts: (1) establishing an upper baseline using French queries, (2) establishing a lower baseline using a known CLIR technique with English queries (probabilistic

[5] http://wordnet.princeton.edu/w3wn.html

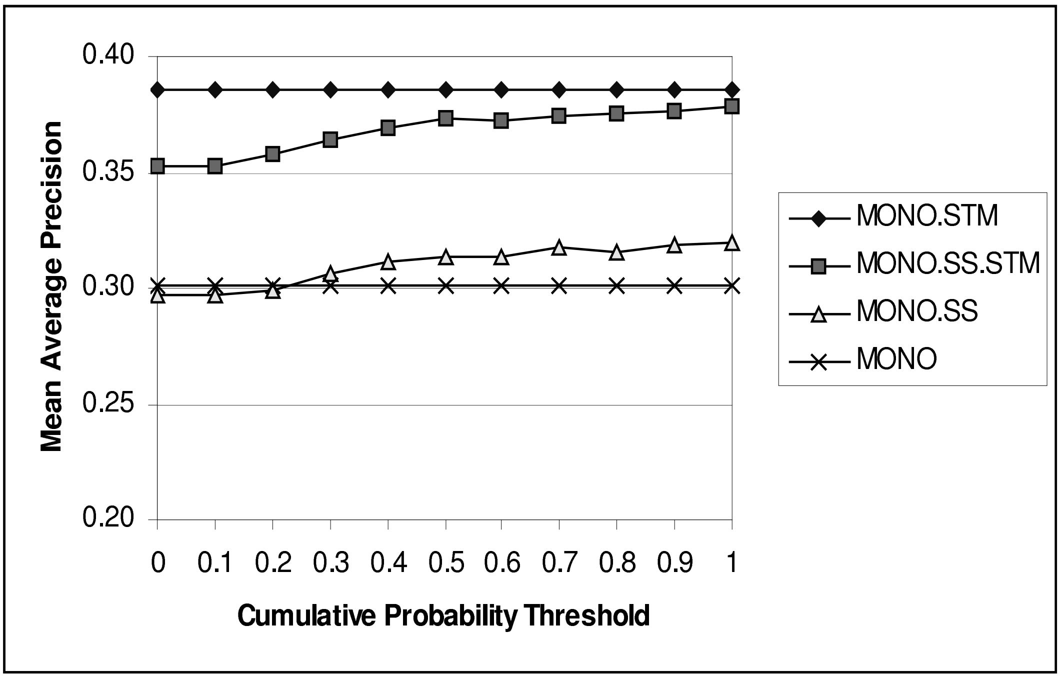


Figure 4.1: Alternative monolingual baselines, TD queries. SS: expansion using statistical synonyms. STM: stemmed using the Snowball French stemmer.

structured queries), and (3) comparing the retrieval effectiveness of the meaning matching model with those baselines. We show meaning matching achieves results statistically indistinguishable from the upper (monolingual) baseline, and significantly better than the lower (CLIR) baseline, while demonstrating less sensitivity to parameter selection.

4.2.1 Upper (Monolingual) Baseline

As we described in Section 2.4.3, it is common to compare CLIR effectiveness with monolingual baseline obtained by retrieving documents with human translated queries in the document language. For this study, we sought to validate our choice of a monolingual baseline by comparing an unexpanded monolingual run with an

Neuchatel	TNO	Hummingbird	Thomason L&R	APL/JHU	BASE	BASE-BRF
0.502	0.488	0.483	0.434	0.392	0.470	0.501

Table 4.1: Comparing with the 5 best official runs of CLEF 2001 monolingual French retrieval. 50 TD queries. Figures are mean average precision. BASE and BASE-BRF are our runs

alternative implementation in which the statistical synonym dictionary was used as a basis for expansion. For each French query, we completed the following steps to construct an expanded query:

- For each query term, find its synonyms based on Equation 3.4.
- Exclude the query term from its synonym list (a term usually appears in its synonym list).
- Re-normalize the probabilities of remaining synonyms.
- Select synonyms based on 11 pre-set Cumulative Probability Thresholds (CPT) ranging from 0 to 1 with an increment of 0.1 at each time.
- Re-normalize the probabilities of selected synonyms, treat them as a pseudo-term and compute the TF and DF of the pseudo-term based on Equation 3.12 and 3.13 and its weight based on Equation 3.14.
- Compute the weight of the original query term. If the term does not have any synonym, double its weight.

- Repeat the above steps for all query terms and adding all the resulting term weights, which will be the document relevance score.

Figure 4.1 shows the monolingual retrieval results when TD queries were used; the results are similar for T queries but with absolute values consistently lower. Stemming yielded large and statistically significant improvements in Mean (uninterpolated) Average Precision (MAP). Similar effects were seen in every CLIR condition, so all CLIR results presented in the figures and tables in this chapter incorporate stemming. Expansion using statistical synonyms yielded a statistically significant 6% relative improvement in MAP when all synonyms were used in the unstemmed condition. No improvement resulted from expansion using statistical synonyms in the stemmed condition. Inspection of the synsets in the unstemmed condition suggests that some of the beneficial effect results from learned synonymy relationships between words sharing a common stem; Such improvement is not possible in the stemmed condition. We, therefore, used the stemmed unexpanded condition as the upper (monolingual) baseline to which we would compare our CLIR techniques (MAP for the stemmed unexpanded monolingual condition is 0.386).

To compare our monolingual baseline with other systems, we computed the MAP over 50 TD queries formulated from the CLEF 2001 topic set (Topics 41-90). Table 4.1 shows the MAP of the top five official monolingual French runs from CLEF 2001. Our monolingual baseline (BASE in Table 4.1) achieved a MAP of 0.470, which is above the average (0.460) of these top five runs but lower than the top three runs. We noticed the best CLEF 2001 run tweaked the stopword list and stemming, and, in particular, used query expansion based on blind relevance feedback [78]. To facilitate comparison, we also expanded our original French queries with the top 20 words selected from the top ten retrieved documents based on Okapi weights, and reduced the weights of added words with a coefficient of 0.1. This resulted in a monolingual MAP of 0.501 (BASE-BRF in Table 4.1) that closely matched the best official run in CLEF 2001 monolingual French retrieval. This suggests that our monolingual baseline is strong. With a goal to study the relative effectiveness of different ways of using translation and synonymy knowledge, we want to avoid masking those effects by other factors. Therefore, blind relevance feedback was not used in the remaining runs.

4.2.2 Lower (CLIR) Baseline: Reexamining PSQ

One of our main goals in developing meaning matching model is to improve over probabilistic structured queries (PSQ) technique. Therefore, we first examine the effectiveness of PSQ, with which only translation knowledge from the query language to the document language is used.

Figure 4.2 shows the effectiveness of PSQ at different CPT values. Notice that the vertical axis denotes the percentage of CLIR MAP over the monolingual baseline (so a value of 100% means CLIR and monolingual retrieval had the same effectiveness). In all the figures of experiments reported in this chapter and the next chapter, we follow the same convention of using percentage monolingual effectiveness. Actual MAP of each CLIR run can be found in Table 4.2. For comparison

CPT	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99	0.999	1
PSQ	0.351	0.359	0.359	0.363	0.364	**0.366**	0.365	0.365	0.363	0.355	0.348	0.328	0.1586
APSQ	0.316	0.315	0.316	0.318	0.318	0.311	0.321	**0.322**	0.318	0.316	0.315	0.272	N/A
PDT	0.353	0.353	0.352	0.353	0.361	0.362	0.362	0.367	0.368	**0.374**	0.369	0.348	0.1699
APDT	0.348	0.348	0.347	0.351	0.359	0.358	0.358	**0.358**	0.358	0.358	0.354	0.336	N/A
APDT.WN	0.362	0.360	0.360	0.360	0.364	**0.365**	0.364	0.357	0.352	0.328	0.219	0.170	N/A
IMM	0.354	0.354	0.354	0.354	0.355	0.359	0.362	0.369	0.374	**0.376**	0.156	0.156	N/A
PAMM-q	0.355	0.355	0.355	0.356	0.356	0.365	0.365	0.372	0.378	0.382	**0.386**	0.368	N/A
PAMM-d	0.353	0.353	0.353	0.354	0.359	0.361	0.369	0.373	0.380	0.384	**0.386**	0.368	N/A
PAMM-d.WN	0.355	0.355	0.354	0.354	0.355	0.361	0.366	0.368	0.372	0.374	**0.378**	0.364	N/A
DAMM	0.351	0.351	0.351	0.356	0.362	0.368	0.375	0.381	0.385	**0.387**	0.257	0.256	N/A
FAMM	0.094	0.099	0.098	0.108	0.113	0.118	0.119	0.121	0.122	0.123	**0.124**	0.124	0.124

Table 4.2:

CLIR mean average precision for different variants of meaning matching model. CLEF test collection: 151 English TD queries, 87,191 French documents. PSQ: Probabilistic Structured Queries; APSQ: Aggregated Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation; APDT.WN: Aggregated Probabilistic Document Translation (aggregation with WordNet); IMM: Individual Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; PAMM-d.WN: Partial Aggregated Meaning Matching in document language (aggregated using WordNet synsets); DAMM: Disconnected Aggregated Meaning Matching; FAMM: Full Aggregated Meaning Matching; N/A: CLIR under this condition was not conducted since retrieval effectiveness already began to decrease obviously.

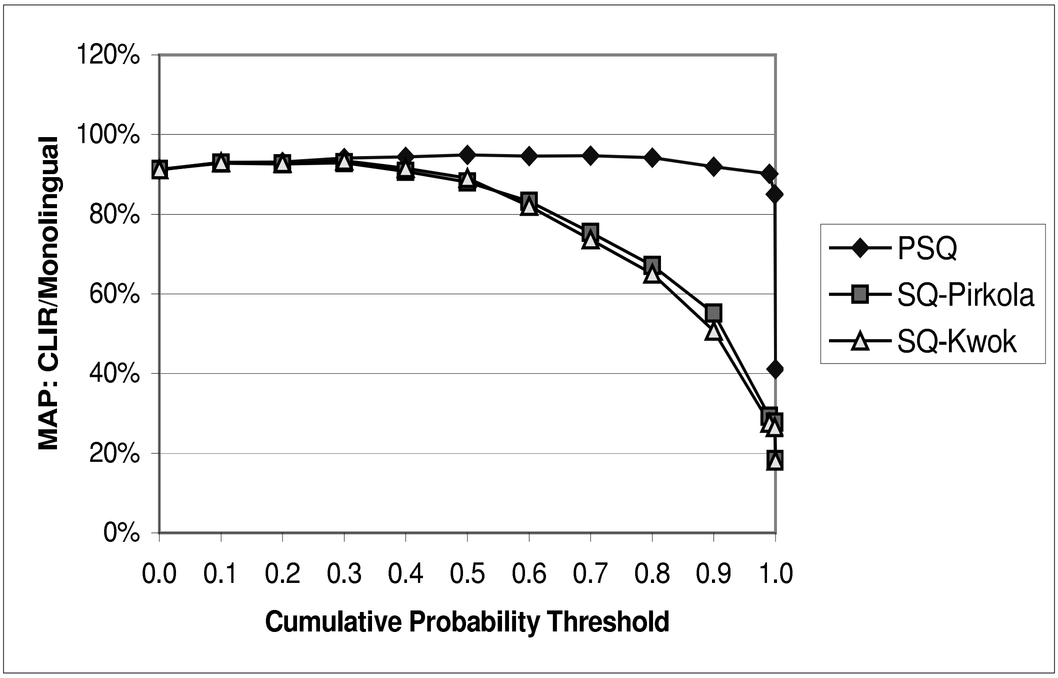


Figure 4.2: Comparison of PSQ and SQ (CLEF). SQ-Pirkola: Pirkola's Structured Queries; SQ-Kwok: Kwok's Structured Queries; PSQ: Probabilistic Structured Queries.

purpose, results of Structured Queries (SQ) are also listed. Notice that the effectiveness of SQ increased as CPT increased, reached its peak near a CPT of 0.3 (MAP of 0.360), and then began to drop dramatically. This peak effectiveness of SQ was significantly worse than the monolingual effectiveness. These findings are consistent with what was reported by Darwish and Oard [15]. PSQ showed more robustness and better effectiveness. The effectiveness of PSQ continued to increase after a CPT of 0.3, and reached it peak near a CPT of 0.5 (MAP of 0.366). It kept relatively stable until near a CPT of 0.9 where it remained at least as good as at a CPT of 0, which corresponds to one-best translation. This is also consistent with the original findings by Darwish and Oard. After a CPT of 0.9 and before a CPT of 0.999, the effectiveness of PSQ dropped slight below one-best translation. However, at a CPT of 1, when all translation alternatives were used, CLIR effectiveness degraded to about 40% of one-best performance (MAP of 0.159). This is quite different from Darwish's findings, because the MAP of PSQ never dropped below one-best translation regardless of a CPT value in his study [15]. PSQ at best (i.e., 0.366 at a CPT of 0.5) achieved about 95% of monolingual effectiveness. However, it was still significantly worse than the latter and indistinguishable from the best of SQ.

As showed in Table 4.2 and 4.3, we used equal increments of 0.1 at lower CPT values as CLIR effectiveness did not show dramatic changes in that region. At high CPT values (e.g., after 0.9), we tried more points of CPT such as 0.99, 0.999, and even 0.9995 to learn where the effectiveness began to degrade. This way, we may give more detailed illustration of the relationship between CLIR effectiveness based on different meaning matching models and the amount of translation knowledge used.

Figure 4.2 shows the results of SQ with Kwok's approximation (which uses the summation of the DF's of translation alternatives to estimate the DF of the query term). CLIR effectiveness was similar between Kwok's SQ and Pirkola's SQ (which uses the size of the union of the documents that contain each translation alternatives) over the region of maximal retrieval effectiveness. This confirms it is appropriate to use Kwok's approximation in place of Pirkola's SQ when meaning matching probability is not used to compute term weight. In the rest of this study, all reported SQ results were obtained with Kwok's approximation.

One possible cause of the slight difference between Kwok's SQ and Pirkola's

SQ involves the type of translation alternatives included at different CPT values. At very low CPT values, usually only one translation was selected, so these two techniques didn't exhibit a noticeable difference. At mid-range CPT values, there was some difference because more than one translation (but not many) were usually used, and some translations might appear together in the same documents. As a result, the summation of the number of the documents containing these translation alternatives might be somewhat larger than the size of these documents' union. At very high CPT values, the number of translation alternatives included became very large. It is likely some of these translation alternatives were common terms with a very high DF that would dominate the effect of other translations on the estimate of DF of a query term, regardless of whether Kwok's SQ or Pirkola's SQ was used. Little difference is therefore seen at high CPT values.

4.2.3 Using Document Translation Knowledge: PDT

The previous section described the results of the simplest situation of meaning matching, in which only translation knowledge from query language to document language was used. Another similar case is to use only translation knowledge in the other direction, from document language to query language. As we discussed in Chapter 3, the meaning matching model developed to handle this situation is called Probabilistic Document Translation (PDT).

Figure 4.3 shows CLIR results with PDT. For comparison purposes, we also include the results with PSQ. Similar to PSQ, the effectiveness of PDT increased as

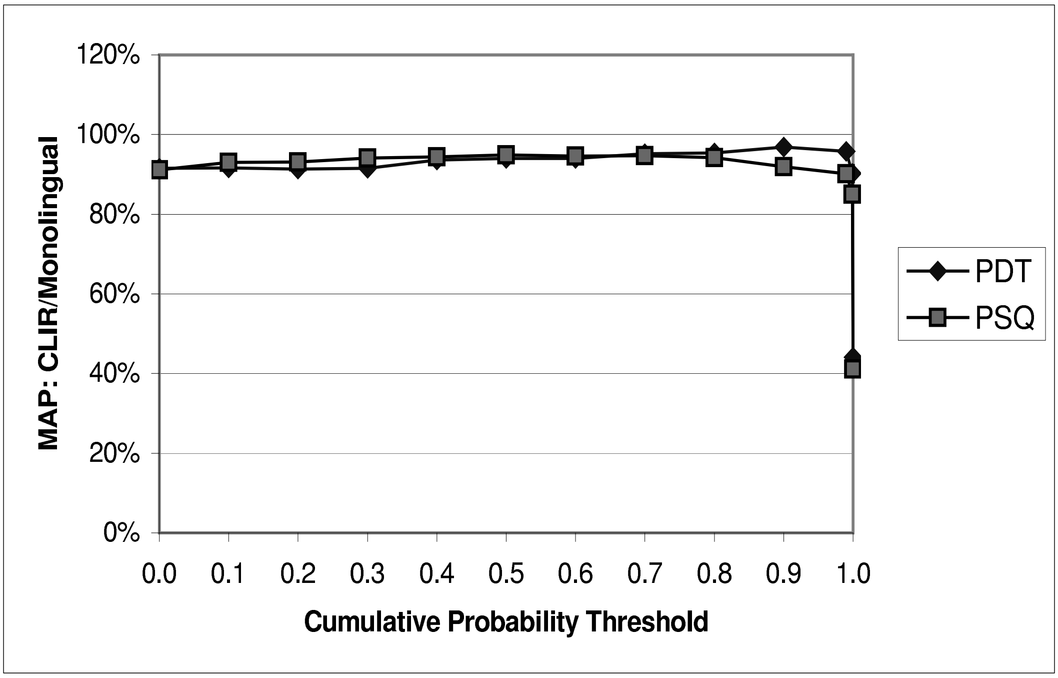


Figure 4.3: Comparison of PSQ and PDT (CLEF). PSQ: Probabilistic Structured Queries; PDT: Probabilistic Document Translation.

CPT increased, reached its peak near a CPT of 0.9 (MAP of 0.374), and dropped sharply below the one-best translation performance when all translation alternatives were used. Compared to PSQ, it seems PDT was slightly worse at lower CPT values, when fewer translation alternatives were used. However, it showed more improvement at higher CPT values. This seems to indicate that PDT is more robust to translation noise. A Wilcoxon signed rank test shows the best of PDT - about 97% of monolingual effectiveness - is indistinguishable from either the best of PSQ or from the monolingual baseline.

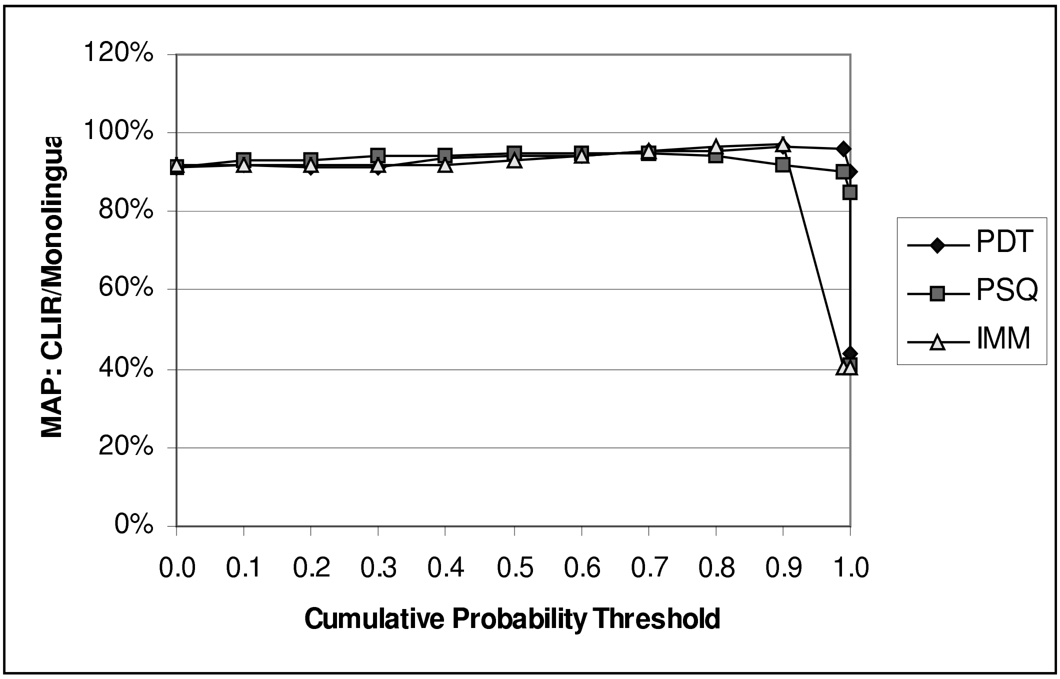


Figure 4.4: Comparison of IMM, PSQ, and PDT (CLEF). IMM: individual meaning matching; PSQ: probabilistic structured queries

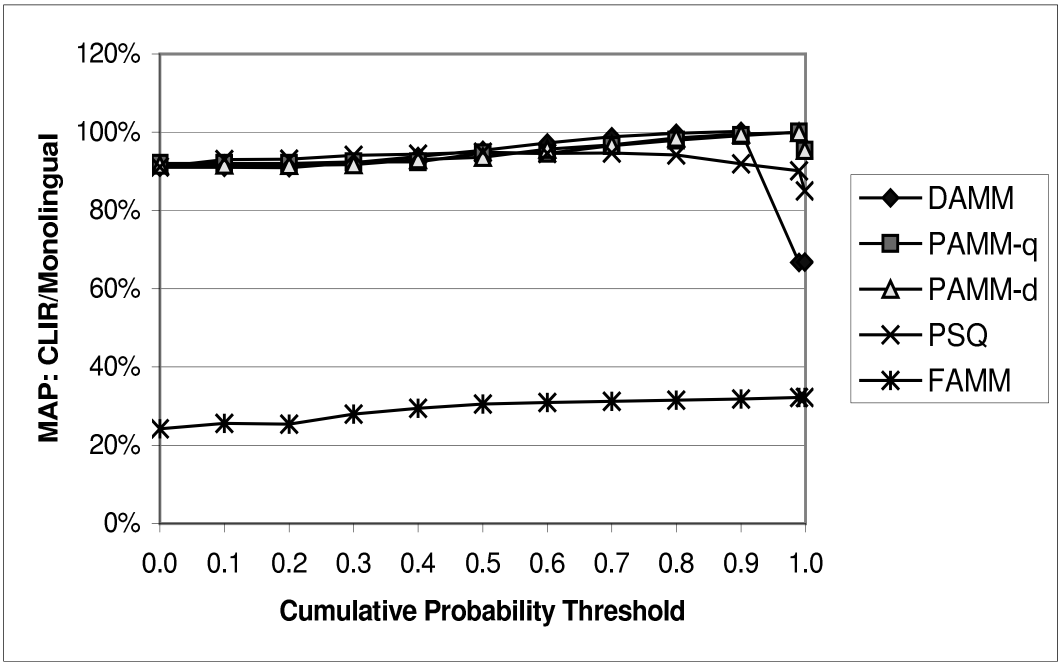
4.2.4 Using Bidirectional Translation Knowledge: IMM

Among all the variants of meaning matching that use bidirectional translation knowledge, IMM is the simplest one because it does not involve synonymy knowledge. The effectiveness of CLIR based on IMM showed monotonic increase before CPT reaches 0.9. The highest MAP (0.376 at a CPT of 0.9), however, is indistinguishable from either the best of PSQ or monolingual effectiveness (see Figure 4.4).

The monotonic increase of MAP at low and medium CPT regions seems to indicate some advantage of using bidirectional translation knowledge over unidirectional translation knowledge. Essentially this is because using bidirectional translation knowledge can both eliminate some spurious translation alternatives that are otherwise included unidirectional translation and gives better estimation of meaning matching probability. However, such effects are quite limited, especially when many low probability translations are included. In fact, after a CPT of 0.9, IMM decreased faster than PSQ and PDT, showing combining bidirectional translation knowledge may have included more *low-probability* translations than using unidirectional translation knowledge. We suspect it is because in a statistical translation model every word can be translate to every word appearing in aligned sentences. We show below that synonymy knowledge can partially offset the negative effect due to the inclusion of too many low-probability translations. In addition, pruning the raw statistical translation models can also reduced the number of spurious translations in IMM.

4.2.5 Combining Bidirectional Translation Knowledge and Synonymy Knowledge: FAMM, PAMM, and DAMM

Figure 4.5 shows the results of matching meaning when both translation knowledge and statistical synonymy knowledge were used. Similar to IMM, PAMM and DAMM showed monotonic improvement of MAP before reaching a CPT of 0.9. What is different is that the effectiveness of PAMM-q and PAMM-d remained better than one-best translation even when all the meaning matching candidates were used. Notice that the best CLIR effectiveness of each of the three variants (DAMM, PAMM-q, and PAMM-d) was comparable to monolingual effectiveness, and they are statistically indistinguishable ffrom each other. The results seem to indicate that



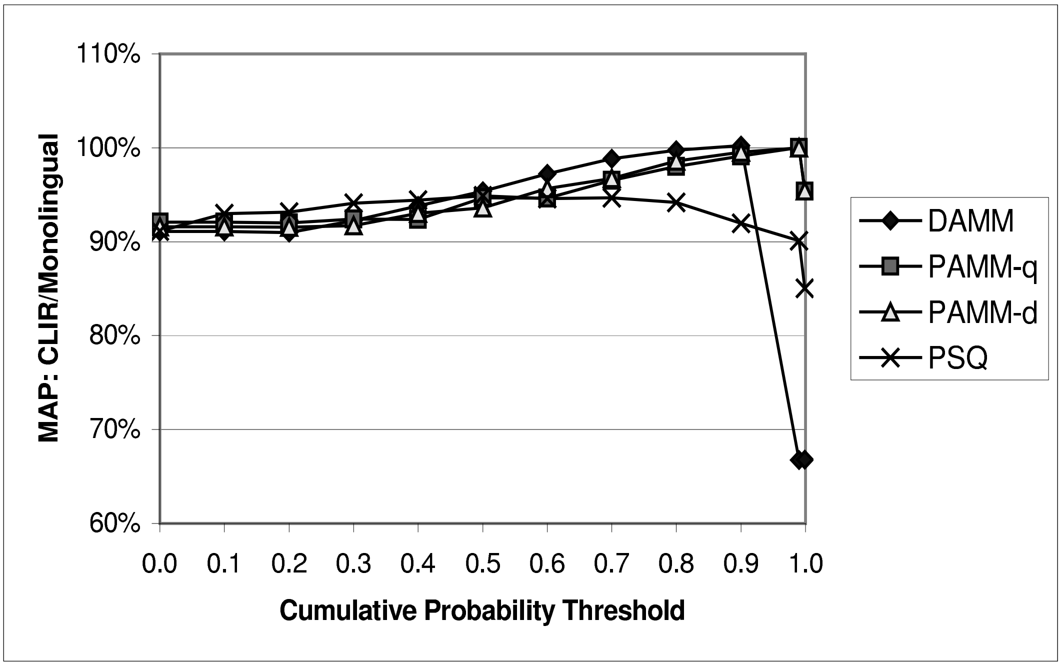


Figure 4.5: Comparison of DAMM, PAMM, FAMM, and PSQ (CLEF). DAMM: Disconnected Aggregated Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; IMM: Individual Meaning Matching; FAMM: Full Aggregated Meaning Matching. Lower figure is a blown-out version of upper figure.

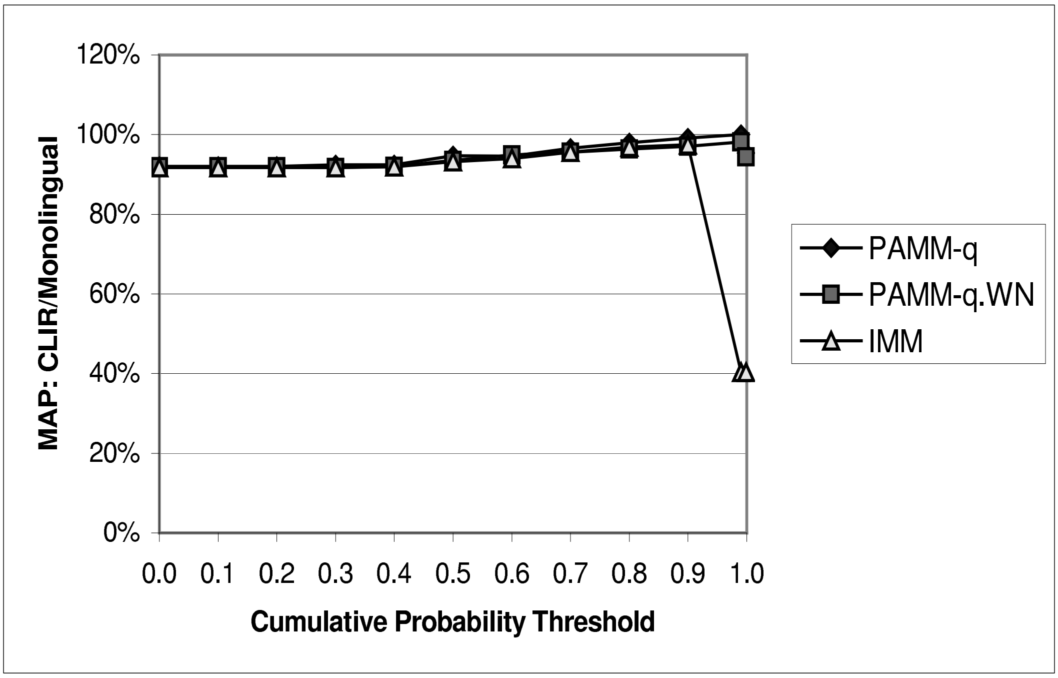


Figure 4.6: Comparison of meaning matching using statistical synonyms and WordNet synonyms (CLEF). PAMM-q: Partial Aggregated Meaning Matching in query language (aggregated using statistical synonyms); PAMM-q.WN: Partial Aggregated Meaning Matching in query language (aggregated using WordNet synonyms); IMM: Individual Meaning Matching.

using bidirectional translation knowledge together with statistical synonymy knowledge has some advantage over using bidirectional knowledge alone. However, the quicker drop of MAP for DAMM after a CPT of 0.99 seems to indicate overuse of statistical synonymy knowledge may have a negative effect when lots of low-probability translations are present.

Figure 4.6 shows the comparison of aggregation in the query language (English) using WordNet synonyms and statistical synonyms. It seems statistical synonyms helped a little bit more than WordNet synonyms, and both had some advantage over IMM, which does not use synonymy knowledge.

Full meaning matching with aligned synsets obtained from EuroWordNet performed significantly worse in every case than other meaning matching that uses statistical synonyms. We found that many high-probability translations contained in the GIZA++ tables were not covered by the aligned synsets. As a result, full aggregated meaning matching treated their probabilities as zero. This is clearly undesirable, and future work on compensating for limited word coverage of aligned synsets is needed.

Overall, aggregation had little effect at low CPT values. The number of translation alternatives included at low CPT values was very small (in most cases there was just one translation selected). Generally, the more translations involved, the larger effect aggregation is likely to have. Therefore, at high CPT values where more translations are included, aggregation tends to have more effect on meaning matching.

Wilcoxon signed rank tests show the best of each of DAMM, PAMM-q, and

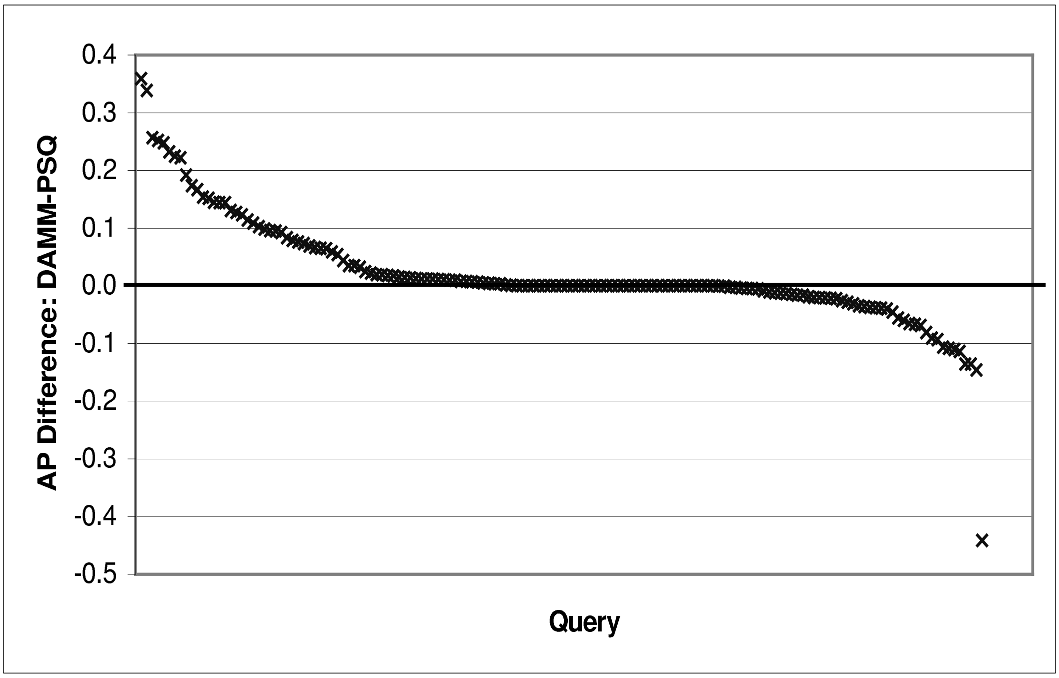


Figure 4.7: Query-by-query comparison of the best DAMM and the best PSQ (CLEF). AP: (uninterpolated) average precision; DAMM: Disconnected Aggregated Meaning Matching; PSQ: probabilistic structured queries.

PAMM-d significantly outperformed the best of PSQ. To further investigate what actually happened, we plot the uninterpolated Average Precision (AP) difference for each query between the best case of DAMM and the best case of PSQ (see Figure 4.7). Among the 151 queries, 67 had higher AP with DAMM, 48 had higher AP with PSQ, and the remaining 36 were the same — revealing the difference between DAMM and PSQ was not due to a small set of topics. The last point in the figure seems to be an outlier. It corresponds to Topic 105, for which only two relevant documents exist in the collection. The two relevant documents appeared at rank 1 and 4 for PSQ, and rank 2 and 17 for DAMM. AP is known to be sensitive to topics with few relevant documents, which explains the large observed difference between DAMM and PSQ for this topic. We did the same comparative analysis for other cases in which a statistically significant difference was observed, and consistently found the difference was not due to a small set of extreme topics.

4.2.6 Combining Unidirectional Translation Knowledge with Synonymy Knowledge: APSQ and APDT

When translation knowledge is available in only one direction, it can also be combined with synonymy knowledge in order to perform meaning matching. The situation is depicted by APSQ and APDT. As Figure 4.8 shows, synonymy knowledge did not work well with unidirectional translation knowledge. In the case of query translation, aggregation significantly degraded CLIR effectiveness at every CPT value. In the case of document translation, aggregation with statistical synonyms never

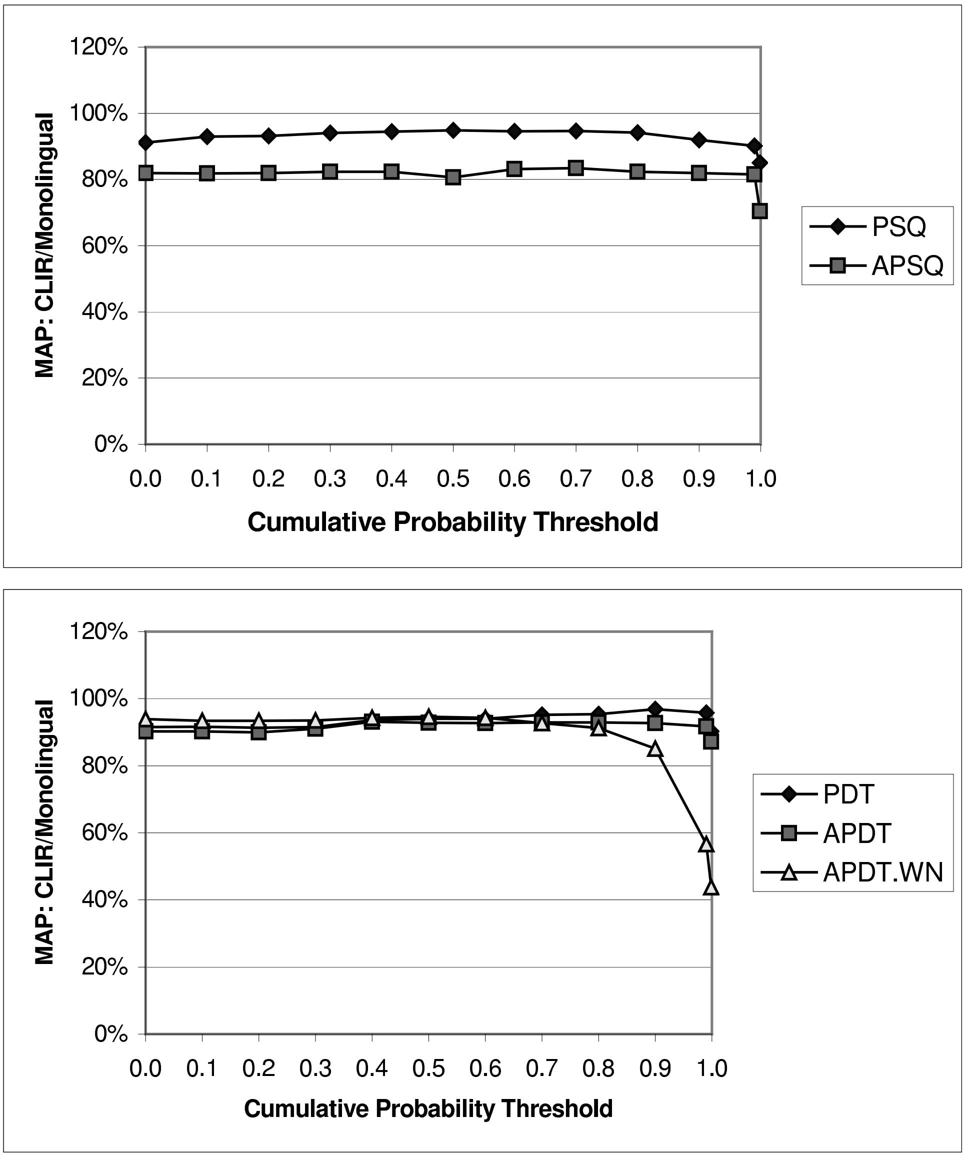


Figure 4.8: Meaning matching: combining unidirectional translation knowledge with with synonymy knowledge. Upper figure: aggregation on query language side. PSQ: Probabilistic Structured Queries; APSQ: Aggregated Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation (aggregation with statistical synonyms); APDT.WN: Aggregated Probabilistic Document Translation (aggregation with WordNet synonyms). Lower figure: aggregation on document language side.

helps — it has little effect at lower CPT values, and it hurts at higher CPT values; aggregation with WordNet synonyms gives mixed results - it seems to help at lower CPT values but hurt more than statistical synonyms at higher CPT values.

A possible explanation for the observed negative effect of aggregation at high CPT values is unidirectional translation contains too many spurious translations, especially at higher CPT values. After being aggregated, some of these spurious translations might have been assigned to synsets with high probability, so their weights were overrated. As a result, non-relevant documents that contain these translations were retrieved, hence CLIR effectiveness decreased. By contrast, aggregation helps meaning matching with bidirectional translation knowledge, since bidirectional translation has the effect of eliminating some spurious translations. Furthermore, aggregation with WordNet synonyms seemed to hurt more than statistical synonyms at high CPT values. We compared statistical synonyms and WordNet synonyms, and found there are more morphological variants in statistical synonyms than in WordNet. Aggregating morphological variants may have less effect than aggregating synonyms not sharing stems, because stemming has a similar effect. At higher CPT values, aggregation with WordNet tends to boost the probability of more spurious translations that are not morphological variants than aggregation with statistical synonyms. This explains why aggregation with WordNet performed worse in high CPT region.

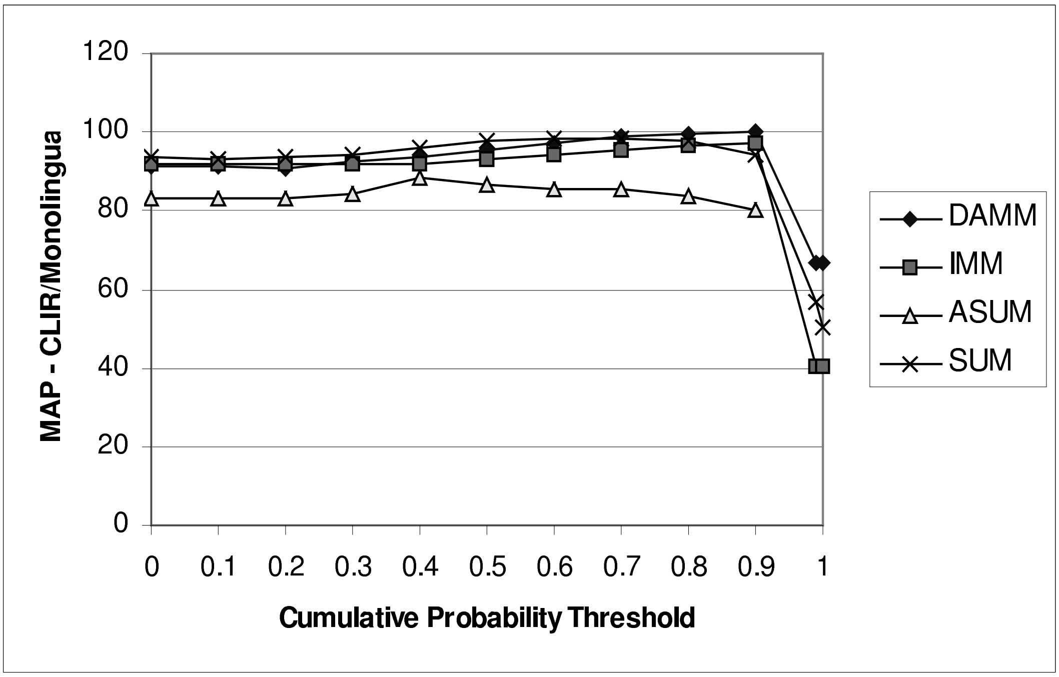


Figure 4.9: Combining bidirectional translation probabilities using multiplication or weighted summation (CLEF). DAMM: Disconnected Aggregated Meaning Matching; IMM: Individual Meaning Matching; SUM: weighted summation of probabilities; ASUM: applying aggregation to weighted summation of probabilities.

4.2.7 Combining Bidirectional Translation Probabilities: Multiplication or Summation?

It is common in CLIR to use weighted summation of translation probabilities to combine translations gained form difference sources. For example, Bayes' rule was applied to convert statistical translations in one direction into translations in the opposite direction. The resulting translations can then be combined with another query-to-document statistical translation model with each receiving equal weight [85, 15]. In our derivation of meaning matching, however, bidirectional translation probabilities are combined in a multiplication basis. We are interested to compare these two methods of using bidirectional translation knowledge.

Figure 4.9 shows the comparison results. Overall, weighted summation of translation probability showed a pattern of change in CLIR effectiveness similar to PSQ. Specifically, it increased as CPT increased at lower values, reached the optimal point somewhere in the medium CPT region, then decreased as CPT moved to higher values. The best effectiveness of weighted summation (0.380 at a CPT of 0.6) was at least as good as the best as IMM (i.e., when synonymy knowledge was not considered.) The Wilcoxon sign rank test reveals that it is significantly better than the best of PSQ, and indistinguishable from all four meaning matching variants in which bidirectional translation knowledge is used (IMM, PAMM-q, PAMM-d, and DAMM). However, adding synonymy knowledge (ASUM) significantly degraded the effectiveness, probably because multiplication tends to "skew" probability distributions among translation alternatives while weighted summation tends to "flatten" probability distributions. As a result, there tend to be more translation alternatives at low CPT values in the weighted summation table than in the multiplication table. As most of these translations have high probabilities, including them could result in better CLIR effectiveness. When moving to high CPT regions, the weighted summation method tends to include more spurious translations than the multiplication, which explains the latter performed better.

In brief, both methods of combining bidirectional translation knowledge can lead to improvement of CLIR effectiveness instead of using unidirectional translation knowledge. The weighted summation method performed better in low CPT regions, while the meaning matching model performed better in high CPT regions. More importantly, when combined with synonymy knowledge, meaning matching performed better.

4.2.8 Using More Accurate Translation Models

The robustness of the meaning matching models can be influenced by the inclusion of too many spurious translations. Although the meaning matching model has the positive effect of eliminating some spurious translation alternatives, at least that ability is limited when the number of translation alternatives becomes large. A simple way to mitigate this risk involves the exclusion of potentially spurious translations in the translation models obtained through statistical MT training before they are used to derive the meaning matching model. In this experiment, we did so by excluding translations beyond cumulative probability threshold of 0.9 from the two raw GIZA++ translation tables. We then derived different variants of meaning matching model from these two reduced translation models.

Table 4.3 shows the mean average precision under different conditions of cross-language meaning matching based on the two reduced translation tables. The same data is displayed in Figure 4.10 except PAMM-d is not listed as it is almost identical to PAMM-q. The best result of each meaning matching variant and the comparison are listed in Table 4.4. Basically, all variants of meaning matching showed similar pattern of changes of CLIR effectiveness. Specifically, CLIR effectiveness increased at lower CPT values, then reached peak performance, and finally decreased at high

CPT	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
PSQ	0.351	0.359	0.359	0.363	0.364	**0.366**	0.365	0.365	0.363	0.355	N/A
PDT	0.353	0.353	0.352	0.353	0.361	0.362	0.362	0.367	0.368	**0.374**	N/A
IMM	0.354	0.354	0.354	0.354	0.355	0.359	0.363	0.368	0.373	**0.375**	0.358
PAMM-q	0.353	0.353	0.353	0.353	0.358	0.361	0.366	0.373	0.378	**0.380**	0.357
PAMM-d	0.356	0.356	0.356	0.357	0.356	0.365	0.365	0.371	0.378	**0.380**	0.358
DAMM	0.359	0.359	0.359	0.364	0.369	0.373	0.378	0.381	0.386	**0.388**	0.348

Table 4.3: CLIR mean average precision for different variants of meaning matching model. CLEF test collection: 151 English TD queries, 87,191 French documents. Original statistical translation models were cut off at a cumulative probability threshold of 0.9. PSQ: Probabilistic Structured Queries; PDT: Probabilistic Document Translation; IMM: Individual Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; DAMM: Disconnected Aggregated Meaning Matching.

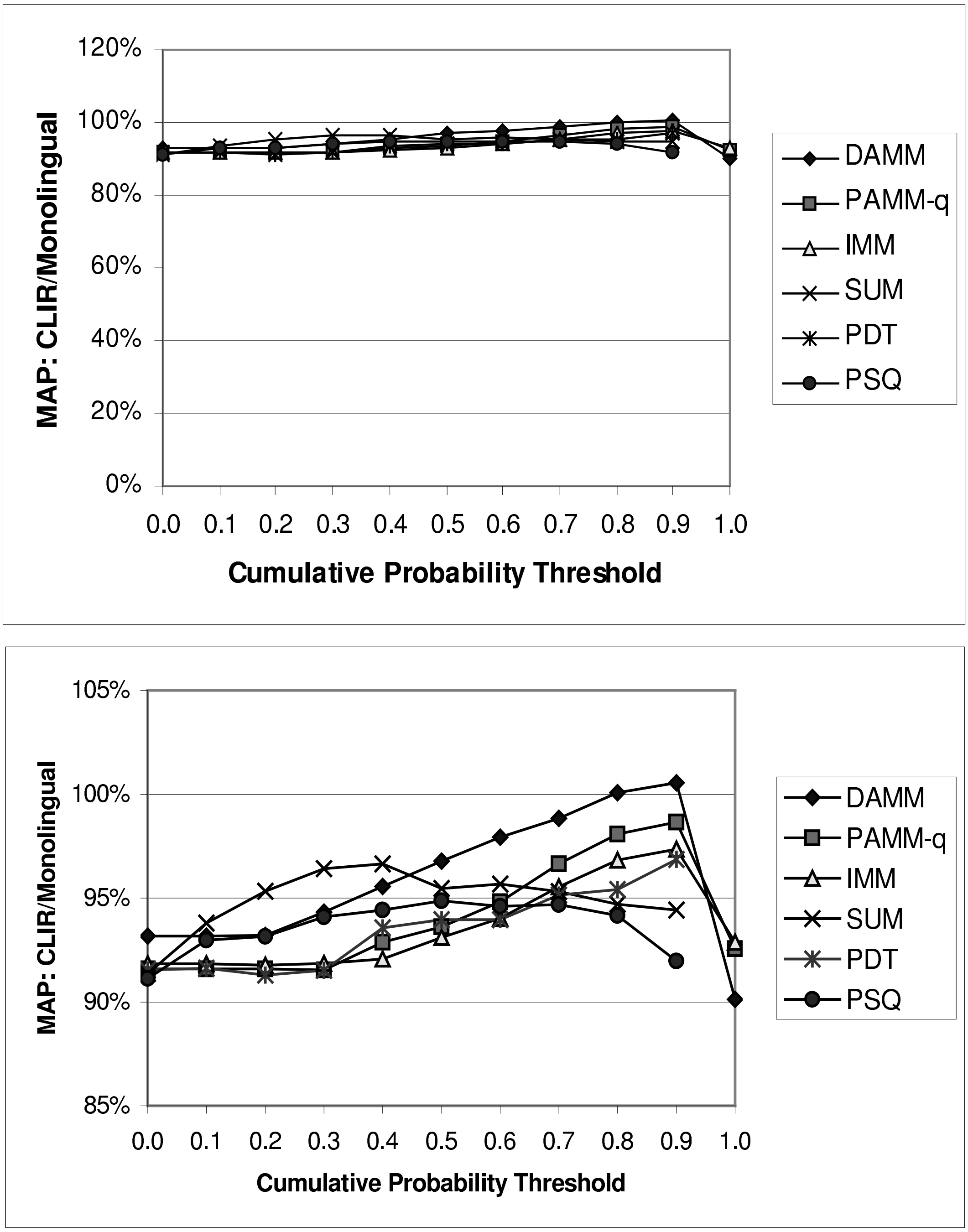


Figure 4.10: Meaning matching based on statistical translation models pruned at cumulative probability threshold of 0.9 (CLEF). The lower figure is a blown-out version of the upper figure.

	PSQ	PDT	SUM	IMM	PAMM-q	PAMM-d	DAMM
MAP (absolute)	0.366	0.374	0.369	0.375	0.380	0.380	0.388
MAP (CLIR/MONO)	95%	97%	96%	97%	99%	99%	101%
$\approx$ MONO?	No	Yes	Yes	Yes	Yes	Yes	Yes
Beats PSQ?	N/A	No	No	No	Yes	Yes	Yes

Table 4.4: Best CLIR mean average precision for different conditions of meaning matching CLEF test collection: 151 English TD queries, 87,191 French documents. Original statistical translations were pruned at a cumulative probability threshold of 0.9.

CPT values. Among all the variants of the meaning matching model, the best CLIR effectiveness was achieved by DAMM (combines translations in both directions and aggregating on both sides). Except for PSQ, which was worse than monolingual effectiveness, all other variants were statistically indistinguishable from monolingual retrieval. Combining bidirectional translation knowledge and statistical synonymy knowledge (DAMM, PAMM-q, and PAMM-d) led to significant improvement of CLIR effectiveness over PSQ, which only used query translation knowledge. However, no statistical significance was observed when bidirectional translation knowledge was not enhanced with synonymy knowledge (IMM).

Comparing the results in Table 4.3 (which pruned GIZA++ translations at a CPT of 0.9) with Table 4.2, we observe some differences. First, MAP of all the variants that use bidirectional translation knowledge no longer dropped as steeply as when translations in the two GIZA++ tables were not pruned. In fact, IMM and

PAMMq always performed at least as well as one-best translation regardless of the cumulative probability threshold, DAMM was just slightly worse when all synonyms were used than in the one-best condition, and the degrade was not statistically significant. This indicates, as expected, pruning statistical translation models did improve the effectiveness of the meaning matching model.

4.3 Summary

Our CLIR experiments with the CLEF English-French collections showed the best retrieval effectiveness of the cross-language meaning matching model (which combines bidirectional translation knowledge and statistical synonymy knowledge), significantly outperformed the best effectiveness obtained using the probabilistic structured query method, achieving CLIR effectiveness comparable to the effectiveness of a strong monolingual baseline. The experiments also showed document translation knowledge may be used in a similar way (i.e., probabilistic document translation) as query translation knowledge is used in probabilistic structured queries, achieving CLIR effectiveness comparable to probabilistic structured queries. Combining translation knowledge in two directions by multiplying translation probabilities seemed to perform better as more low probability translation alternatives were included. As with probabilistic structured queries, cross-language meaning matching using statistical translation models can be adversely influenced by large numbers of low-probability translation alternatives. When meaning matching is enhanced with bidirectional translations and synonyms, CLIR effectiveness is more robust to noisy translations. In all the different cases of meaning matching that use bidirectional translation knowledge, CLIR effectiveness at cumulative probability of 0.9 was always significantly better than using the most probable translation alone. In addition, it is effective to apply a cumulative threshold to the raw statistical translation models before the full-fledged meaning matching model is developed.

Overall, English-French CLIR is relatively simple and effective. All variants of cross-language meaning matching achieved at least 95% of monolingual effectiveness. In fact, Even using one-best translation obtained from statistical MT training led to 92% monolingual effectiveness. On the other hand, despite the closeness of their MAP's, we still observed statistical significance by comparing the baseline probabilistic structured query and meaning matching that uses bidirectional translation and synonymy knowledge. All experiment results suggested the usefulness of the cross-language meaning matching model.

Any technique proved effective for one language pair with one test collection is subjected to further tests. In the next chapter, we study the meaning matching model with a different language pair: English-Chinese. Different from French, Chinese presents extra problems such as word segmentation and character encoding conversion but does not have morphology. We want to see what findings we learned from the English-French experiments hold and what findings change for English-Chinese CLIR, and if so, what factors have caused such changes.

Chapter 5

English-Chinese CLIR Experiments

In this chapter, we present our CLIR experiments with English queries to search Chinese documents. We used two test collections, one from TREC-5 and TREC-6, the other from TREC-9. We first evaluated our meaning matching model with TREC-9 collection. The relatively small set of 25 topics makes it difficult to interpret comparative results with statistical significance tests. Therefore, we ran similar experiments with a combined English-Chinese CLIR collection from TREC-5 and TREC-6, which has a larger set of 54 topics.

In addition to the issues investigated in Chapter 4, we also want to examine less well-trained statistical translation models on the performance of the meaning matching model. We will show the meaning matching model consistently outperformed the PSQ CLIR baseline, achieving CLIR effectiveness comparable to a monolingual baseline. In addition, we found that including HMM iterations in statistical MT training had little effect on the effectiveness of meaning matching.

5.1 Experiment Design

This section describes the test collections, query formulation and document processing, statistical translation model training, and statistical synonym identification.

5.1.1 Test Collection and Query/Document Processing

TREC-9 “xlingual” (CLIR) test collection contains 126,937 Chinese documents encoded in BIG5, 25 English topics, and pooled relevance judgments. The documents are articles from newspapers published in Hong Kong from 1998 – 1999. We converted the documents from BIG5 into UTF-8 using the uconv code-set conversion tool, then segmented them into space-delimited strings of words using a modified version of the LDC Chinese segmenter.[1] The resulting document collection was then converted into hexadecimal format for indexing. At least one previous study has demonstrated the feasibility of using hexadecimal format in retrieval systems that have troubles handling Chinese characters [45].

The Chinese version of the original topics was encoded in BIG5. We converted them into UTF-8 encoding and segmented them into words with the same segmenter used to process the document collection. Finally, we retained all the words appearing in the title and description fields to formulate TD queries for the monolingual baseline. For topics in English, we removed words that belong to the same stopword list provided with Inquery from the title and description fields and kept the rest to formulate TD queries for CLIR runs.

[1] http://www.ldc.upenn.edu/Projects/Chinese/segmenter/mansegment.perl

The TREC-5 Chinese collection contains 139,801 documents from People's Daily published from 1991 – 1993 and 24,988 documents from Xin Hua New's Agency published in 1994 – 1995. The document collection was re-used in TREC-6. For TREC-5, 28 topics were constructed, and for TREC-6 another 26 topics were created. The topics were supplied in both English and Chinese. We combined these two sets of topics for a total 54 search topics. Documents in GB code were converted into UTF-8 using the same uconv tool and then segmented into individual words using the LDC segmenter. Codeset conversion and segmentation were also performed for formulating Chinese queries. English queries were created in the same way as in TREC-9.

The experiments with both test collections were conducted using PSE, the same IR system used in the English-French CLIR experiments in the previous chapter.

5.1.2 Training Statistical Translation Models

Training of statistical translation models was conducted using the GIZA++ toolkit on the FBIS corpus. The same version of LDC segmenter was used on the Chinese side of the corpus. After removing implausible sentence alignments (token ratio smaller than 0.2 or larger than 5), the resulting 1,583,807 English-Chinese sentence pairs were actually used for MT training.

For the English-French CLIR experiments, we used 10 Model 1 iterations, followed by 5 HMM iterations, and ended with 5 Model 4 iterations. Model 1 iteration is the basic process of producing an alignment. It assumes any word in the corpus of the source language can translate to any word in the target language that co-occurs in aligned sentences. Model 4 uses "fertility" and distortion. Fertility specifies the number of words in the target sentence to which one word in the source sentence may align. In Model 4, fertility could be 0, 1, or larger than 1. Distortion considers a source word more likely to align to some target positions than to other positions. However, Model 4 iteration is more inefficient than Model 1 iteration. In the English-Chinese CLIR experiments, we were interested in how our meaning matching model would perform if Model 4 was not used in statistical MT training. Therefore, we set GIZA++ to output two translation tables - one after finishing 10 IBM Model 1 iterations and the other after finishing 5 HMM iterations. Considering training in both directions, we finally totaled four raw translation tables after completing statistical MT training. For both efficiency and effectiveness concern, we pruned all the four translation models at a cumulative probability threshold of 0.99 before they were used to derive the cross-language meaning matching models.

In the rest of this chapter, we focus on presenting the experiments results based on the two GIZA++ tables that involved only IBM Model 1 iterations. Results based on the translation tables that used both Model 1 and HMM iterations are described in Section 5.2.7.

We identified statistical synonyms based on statistical translation models that involved only Model 1 iterations, and excluded synonyms whose normalized probability is smaller than 0.1.

	APL	MSRCN	IBM	TextWise	FDU	BBN	INQ	BASE
TDN	0.309	0.300	0.297	0.304		0.288		
TD					0.189	0.248	0.268	0.245

Table 5.1: Comparing with top official runs of TREC-9 monolingual Chinese retrieval. TDN: queries with title, description, and narrative field; TD: queries with title and description field. BASE is our run.

5.2 Results

Although we ran our experiments separately with the two test collections, in this section when we examine each variant of the meaning matching model, we present the experiment results with the two collections and our analysis simultaneously. However, it is not our intention to compare absolute retrieval effectiveness such as MAP between the experiments with these two test collections, because it is less meaningful to compare CLIR effectiveness gained from different test collections. We establish a monolingual baseline with each test collection to serve as the "upper-bound" reference for our meaning matching model. We then present results and analysis in a similar way as in Chapter 4.

5.2.1 (Upper) Monolingual Baseline

Monolingual Chinese retrieval (using Chinese queries to retrieve Chinese documents) achieved a mean (uninterpolated) average precision (MAP) of 0.245 with the TREC-9 collection and 0.323 with the TREC-5&6 collection. Both were used TD queries with word-based retrieval. Table 5.1 shows MAP of some top official runs of TREC-9 monolingual Chinese retrieval.[2] Long queries (TDN) often outperform medium queries (TD), as displayed in the table. Among the three official TD runs, the best one (INQ) used query expansion. The second best (BBN) had the closest condition to ours (word-based retrieval, without expansion). BBN later reported that word-based retrieval was not as good as retrieval based on a combination of character bigrams and trigrams [85]. For TREC-5&6 monolingual baseline, we extracted the 19 of the 27 TREC-5 queries that were evaluated by the time when the conference held. We got a MAP of 0.280, which equals to the median of 15 automatic official runs.[3] Most of these official runs used TDN queries and techniques such as query expansion based on blind relevance feedback. Our monolingual baseline was performed with TD queries on word-based index and did not apply other performance-enhancing techniques. Considering these facts, we think our monolingual baseline is a reasonable baseline.

5.2.2 Lower (CLIR) Baseline: PSQ and SQ

CLIR MAP's for TREC-9 and TREC-5&6 are listed in Table 5.2 and Table 5.3 respectively. The two statistical translation models were obtained from running GIZA++ with IBM Model 1 iterations only. The monolingual percentage of probabilistic structured queries (PSQ) and structured queries (SQ) is displayed in Figire 5.1. For TREC-9, somewhat surprisingly, both the best of PSQ (at the CPT of

[2] http://trec.nist.gov/pubs/trec9/appendices/A/xlingual_results.html

[3] http://trec.nist.gov/pubs/trec5/t5_proceedings.html

CPT	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99	0.999	0.9995	1.0
SQ	0.253	0.253	0.251	**0.254**	0.234	0.238	0.226	0.212	0.129	0.057	0.004	N/A	N/A	N/A
PSQ	0.254	0.254	0.253	0.242	0.234	0.251	**0.284**	0.274	0.282	0.278	0.279	N/A	N/A	N/A
APSQ	0.128	0.128	0.133	0.133	0.129	0.154	0.135	0.135	0.135	0.146	**0.156**	N/A	N/A	N/A
PDT	0.226	0.227	0.218	0.226	0.241	0.245	0.276	0.302	**0.308**	0.305	0.172	N/A	N/A	N/A
APDT	**0.278**	0.278	0.278	0.276	0.252	0.239	0.238	0.245	0.275	0.277	0.255	N/A	N/A	N/A
IMM	0.274	0.274	0.274	0.273	0.241	0.250	0.250	0.274	0.267	0.272	0.276	**0.276**	0.265	0.068
PAMM-q	0.259	0.259	0.259	0.245	0.246	0.268	0.268	0.271	0.272	0.273	0.286	**0.287**	0.275	0.089
PAMM-d	0.257	0.257	0.256	0.259	0.250	0.271	0.263	0.280	0.284	0.293	**0.302**	0.302	0.280	0.059
DAMM	0.256	0.258	0.253	0.255	0.268	0.276	0.310	0.305	0.302	0.310	**0.314**	0.314	0.313	0.060

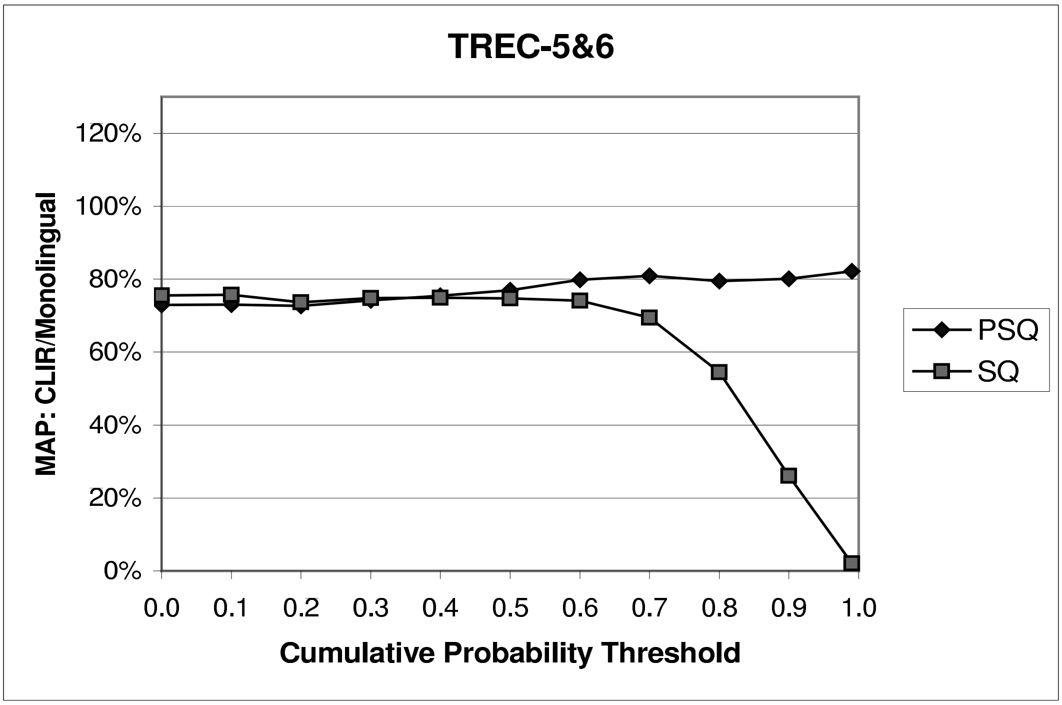
Table 5.2:

CLIR mean average precision for different variants of meaning matching. TREC-9 collection: 25 English TD queries, 127,758 Chinese documents. Statistical translation models trained with 10 IBM Model 1 iterations, translations pruned at the cumulative probability threshold of 0.99. PSQ: Probabilistic Structured Queries; APSQ: Aggregated Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation; IMM: Individual Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; DAMM: Disconnected Aggregated Meaning Matching.

CPT	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99	0.999	1
PSQ	0.235	0.236	0.235	0.239	0.244	0.248	0.258	0.261	0.257	0.259	**0.265**	N/A	N/A
APSQ	0.179	**0.179**	0.173	0.175	0.174	0.172	0.172	0.172	0.172	0.172	0.171	N/A	N/A
PDT	0.223	0.224	0.234	0.240	0.259	0.263	**0.284**	0.282	0.265	0.149	0.014	N/A	N/A
APDT	0.301	0.301	0.302	0.302	0.301	0.307	0.310	**0.312**	0.309	0.312	0.270	N/A	N/A
IMM	0.234	0.234	0.234	0.234	0.238	0.249	0.257	0.264	0.271	0.281	0.29	**0.291**	0.120
PAMM-q	0.242	0.242	0.242	0.241	0.245	0.252	0.26	0.264	0.284	0.292	**0.298**	0.298	0.141
PAMM-d	0.229	0.229	0.230	0.248	0.253	0.270	0.280	0.295	0.306	0.316	0.316	**0.316**	0.109
DAMM	0.242	0.242	0.248	0.274	0.278	0.294	0.301	0.309	0.309	0.314	0.316	**0.316**	0.113

Table 5.3:

CLIR mean average precision for different variants of meaning matching. TREC5&6 collection: 54 English TD queries, 164,789 Chinese documents. Statistical translation models: 10 IBM Model 1 iterations, translations pruned at the cumulative probability threshold of 0.99. PSQ: Probabilistic Structured Queries; APSQ: Aggregated Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation; IMM: Individual Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; DAMM: Disconnected Aggregated Meaning Matching.



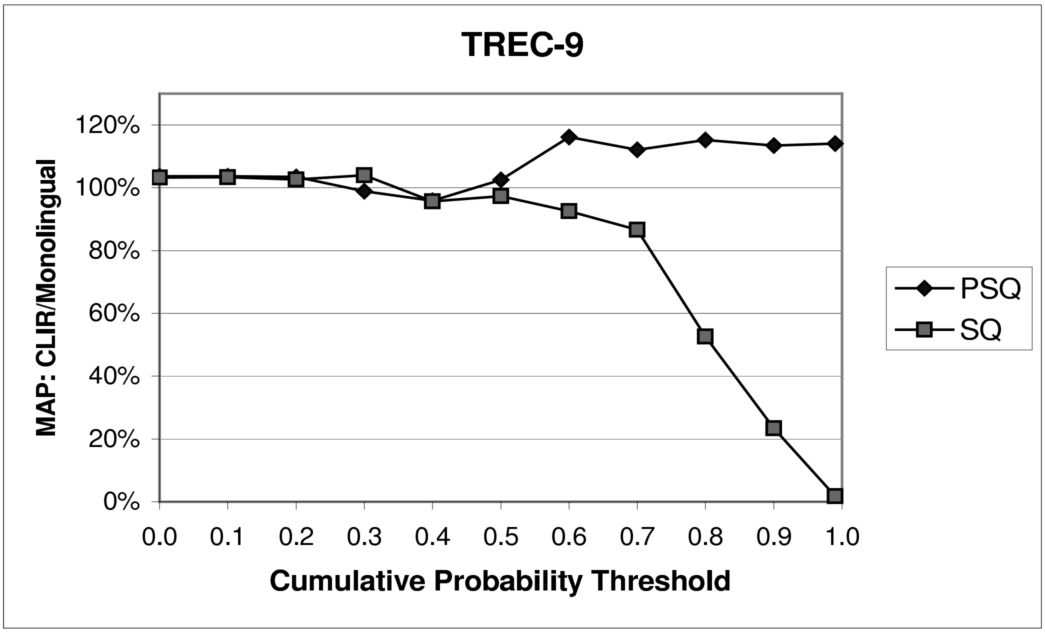


Figure 5.1: Matching meaning with translation knowledge from query language to document language. PSQ: Probabilistic Structured Queries; SQ: Structured Queries.

0.7) and the best of SQ (at the CPT of 0.4) achieved MAP larger than monolingual MAP, although the difference was not statistically significant. For TREC-5&6, the best of PSQ (at the CPT of 0.6) was about 82% of monolingual effectiveness as measured by MAP, while the best of SQ (at the CPT of 0.1) was about 76%. For both TREC-9 and TREC-5&6, PSQ showed a tendency to perform better as CPT increased. In fact, it never dropped below the MAP of one-best translation. By contrast, SQ consistently dropped as more translations were used. However, the Wilcoxon sign rank test shows the best effectiveness of SQ and PSQ are indistinguishable.

Compared to the results for the CLEF English-French CLIR (see Figure 4.2), we found SQ showed the same pattern of changes. However, PSQ exhibited some difference. In the CLEF experiments, PSQ achieved the best MAP in the medium CPT region and after that steadily decreased in the high CPT region. By contrast, in the TREC-5&6 experiments, after reaching peak MAP in the medium CPT region, it remained the same effective even when moving into the high CPT region. This is also true for the TREC-9 experiments. A possible reason for the observed difference of PSQ between the CLEF experiments and the TREC experiments is that at high CPT values, there are more spurious translations in the English-French translation model than in the English-Chinese table. This in turn may be attributed to the different sizes of the parallel corpora used for statistical MT training — the English-Chinese corpus contains about twice more sentences pairs than the English-French corpus. Usually, the bigger the training corpus, the more accurate the statistical MT model. Therefore, the English-Chinese translation model is probably more accurate

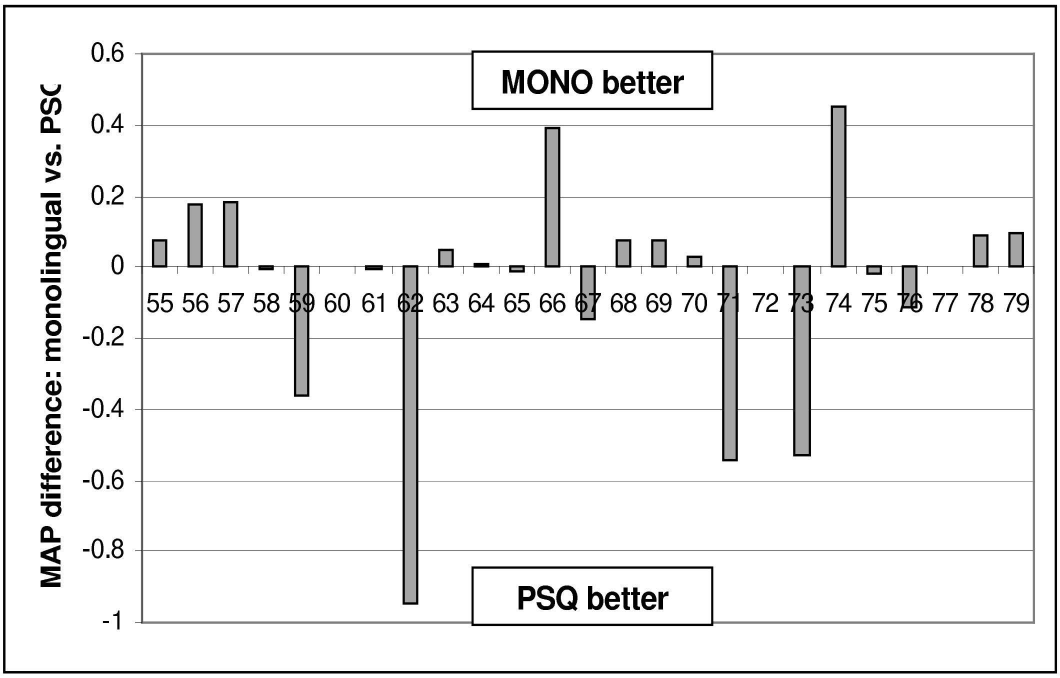


Figure 5.2: Query-by-query comparison of monolingual baseline with the best PSQ (at CPT of 0.6) (TREC-9).

than the English-French translation model.

PSQ performed as well as monolingual retrieval for TREC-9, and it is the first time that we observed such a comparative result in all our studies of PSQ. For this reason, we compared the monolingual baseline with the best of PSQ run (at CPT of 0.6) query-by-query (see Figure 5.2. We found that four queries had much better average precision in CLIR than in monolingual IR, Query 59, 62, 71, and 73 (see Figure 5.2). We compared the translated queries and the Chinese queries and found the following:

- Query 59 ("stealth technology in Asia; what Asian countries are developing stealth ships or aircraft or stealth countermeasures technology"): Translation of the word “stealth” in the Chinese version of the TREC topics is “YinMi”, whereas the most probable translation suggested by the statistical translation model is “YinXing”. Although both “YinXing” and “YinMi” are correct translations of “stealth,” “YinXing” is almost always used as the translation of “stealth” in Mandarin Chinese. Since the documents in the collection are written in Mandarin Chinese, the monolingual query with “YinMi” will almost always hit irrelevant documents, while “YinXing” will almost always retrieve relevant documents. This is evidenced by the average precision of 0 for the manually-translated query and the average precision of 0.273 for the automatically-translated query.

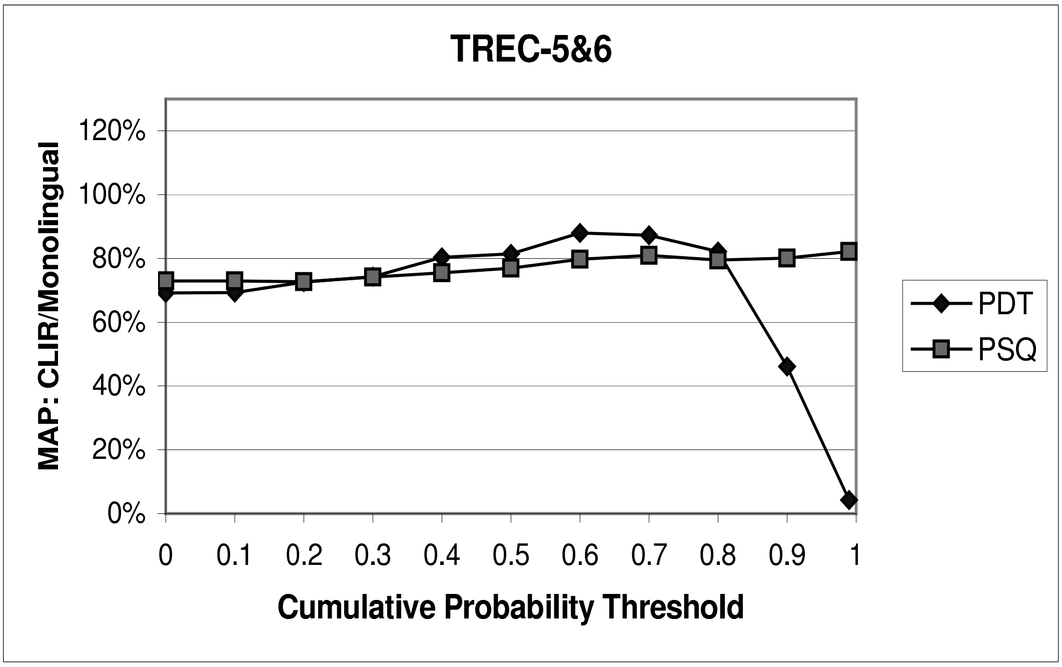
- Query 62 (“Daya Wan nuclear power plant; how much of the electricity generated by the Daya Wan nuclear power plant is sent to Hong Kong”): In the TREC manual translation, “nuclear power plant” is translated into a three-character Chinese word “HeDianChang,” which is a perfect translation. On the other hand, the statistical translation model suggested more useful synonyms of “HeDianChang,” such as “DianZhan” or “FaDianChang.” In fact, when talking about Daya Wan nuclear power plant, “HeDianZhan” is used more often than “HeDianChang.” This explains why the automatically translated query performed better.

- Query 71 (“China and the Olympics; find documents that describe China’s interest in hosting the Olympics”): Manual translation of “Olympics” provided by TREC is a 7-character Chinese word, and after segmentation it becomes two words “AoLinPiKe” and “YunDongHui.” Although both are perfect translation and perfect segmentation, the query with these words can only hit a small number of relevant documents. This is because a three-character acronym “AoYunHui” is more often used in Mandarin Chinese as the translation of “Olympics.” In fact, this is exactly the most probable translation suggested by the statistical translation model.

- Query 73 (“AIDS in China; find documents that report on the number of cases of AIDS in China the names and locations of aids research and treatment facilities in China and the number of deaths per year attributed to AIDS in China”): Translation of “AIDS” is unambiguously a three-character Chinese word “AiZiBing” in both the manual translation and statistical model. However, since word segmentation was conducted on the manual query, “AiZiBing” was incorrectly segmented into three one-character word “Ai,” “Zi,” and “Bing.” “Ai” and “Bing” often mean “love” and “disease” respectively, while “Zi” could have several different meanings. We suspect that “Ai” and “Bing” in the manual query retrieved many non-relevant documents, which accounts for the poor effectiveness.

5.2.3 Using Document Translation Knowledge

Figure 5.5 shows the results of cross-language meaning matching with document translation knowledge alone, probabilistic document translation (PDT). PDT displayed different behavior than PSQ, with effectiveness increasing more in medium



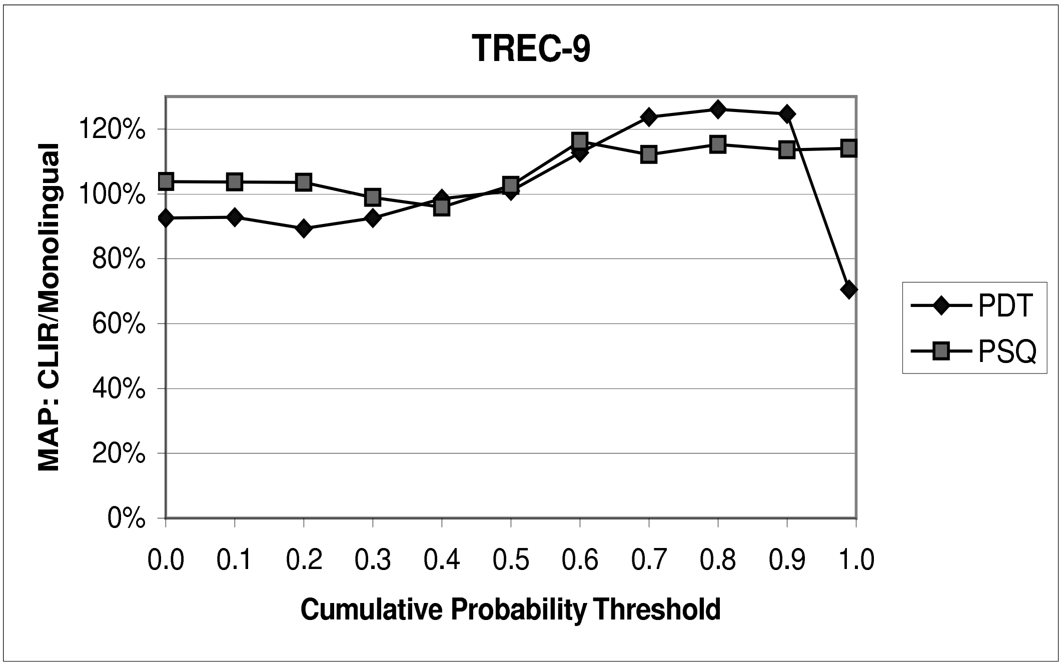


Figure 5.3: Comparison of PDT and PSQ. PDT: probabilistic document translation; PSQ: probabilistic structured queries.

CPT region and dropped more in the high CPT region. With the TREC-9 test collection, the best PDT achieved was about 126% MAP of monolingual IR, while the best PSQ achieved was near 116% MAP of monolingual baseline. With TREC5&6, these two numbers are 88% and 80% respectively. However, the difference between the best of PDT and the best of PSQ is not statistically significant in either case.

A possible reason for PDT to drop more sharply than PSQ in the high CPT region involves the difference of vocabulary size between English and Chinese. In the parallel corpus used for training statistical translation models, there are about 126,000 unique English words and 60,000 unique Chinese words. The size of the English vocabulary is double the size of the Chinese vocabulary. On average each Chinese word has twice the translation alternatives as each English word. As a result, in the high CPT region where the number of low-probability translations increases quickly, PDT that translates from Chinese to English may be affected more than PSQ that translates from English to Chinese. This may also explain the observed difference of PDT between the CLEF experiments and the TREC experiments — the MAP of PDT did not decrease in the CLEF experiments as dramatically as in the TREC experiments after reaching the peak.

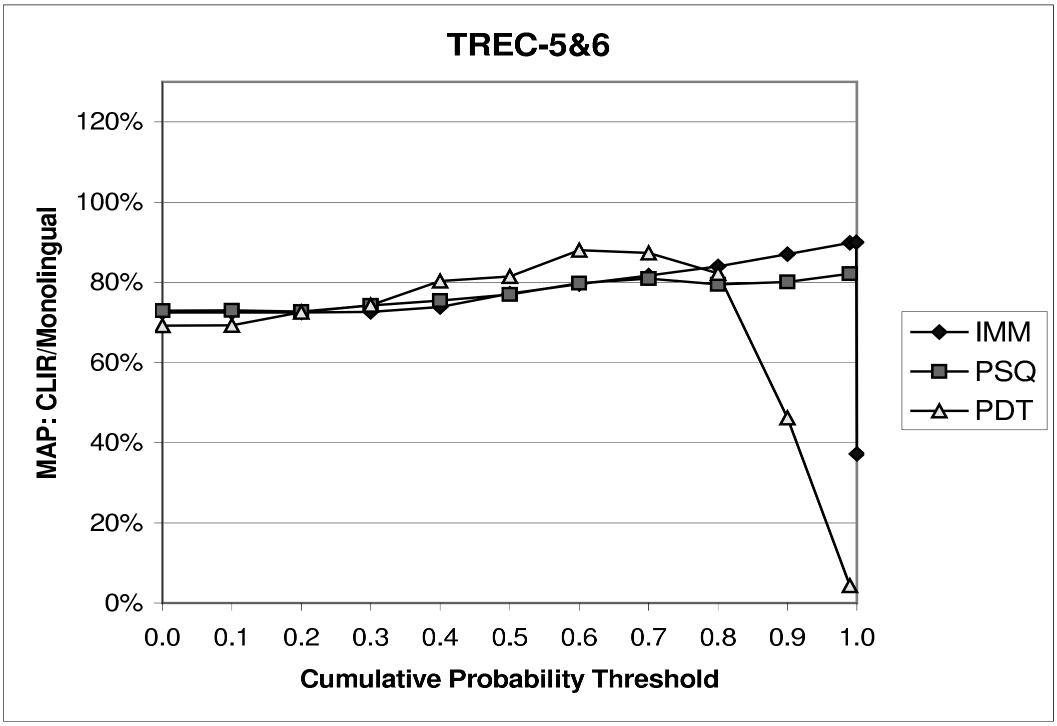
Statistical translation is asymmetric, particularly for language pairs like English and Chinese. IBM Model 1 allows one-to-many alignment but does not allow many-to-one alignment. It is more common for a Chinese word to translate to an English phrase than for an English word to translate to a Chinese phrase. Training an English-Chinese translation model will likely be affected adversely by this restriction from Model 1, whereas training a Chinese-English translation model will probably benefit from it. Therefore, the resulting Chinese-English translation model might be more accurate than the English-Chinese translation model in terms of the number of correct translations included and their translation probabilities. So, document translation may work better than query translation with translation models trained in this way. For this reason, techniques such as "alignment templates" may be helpful, especially for training translation models from English to Chinese.

5.2.4 Using Bidirectional Translation Knowledge: IMM

Using bidirectional translation knowledge (IMM) provided marginal help for improving CLIR effectiveness over using query translation knowledge alone (PSQ). The best MAP of IMM (near the CPT of 0.99 in both TREC-5&6 and TREC-9 experiments) is 123% and 90% of monolingual effectiveness for TREC-9 and TREC-5&6 respectively. The best IMM was significantly better than the best PSQ for TREC-5&6, but statistically indistinguishable from either the best of PSQ for TREC-9. In both experiments, the best IMM is statistically indistinguishable from the best PDT, or the monolingual baseline (see Figure 5.4. Compared to the CLEF results, IMM exhibited a similar pattern of changes (see Figure 4.4).

5.2.5 Combining Bidirectional Translation Knowledge and Synonymy Knowledge: PAMM and DAMM

For both TREC-5&6 and TREC-9, combining bidirectional translation knowledge and synonymy knowledge (PAMM-q, PAMM-d, and DAMM) significantly improved



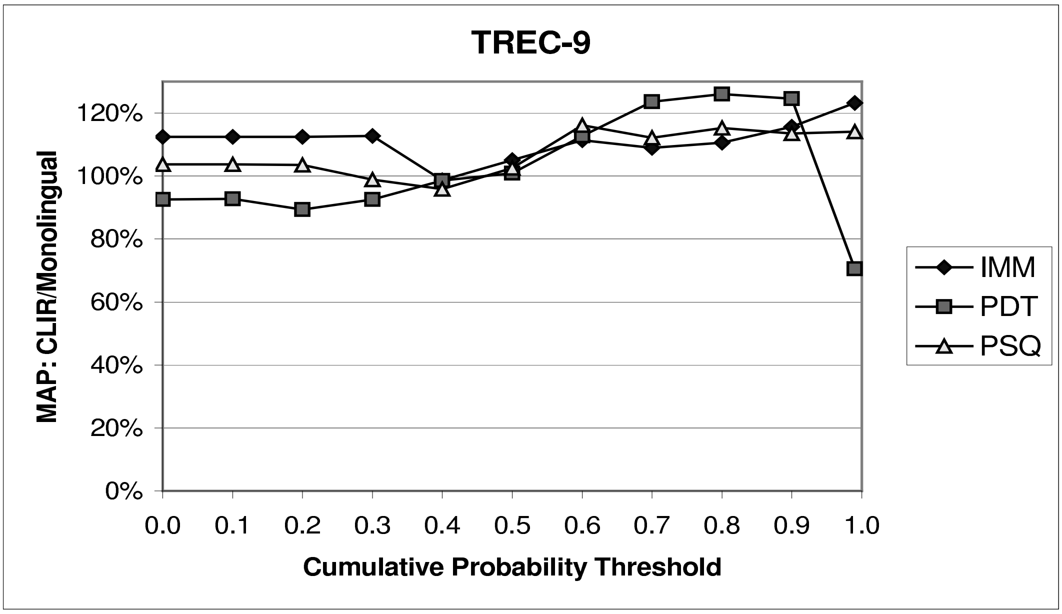
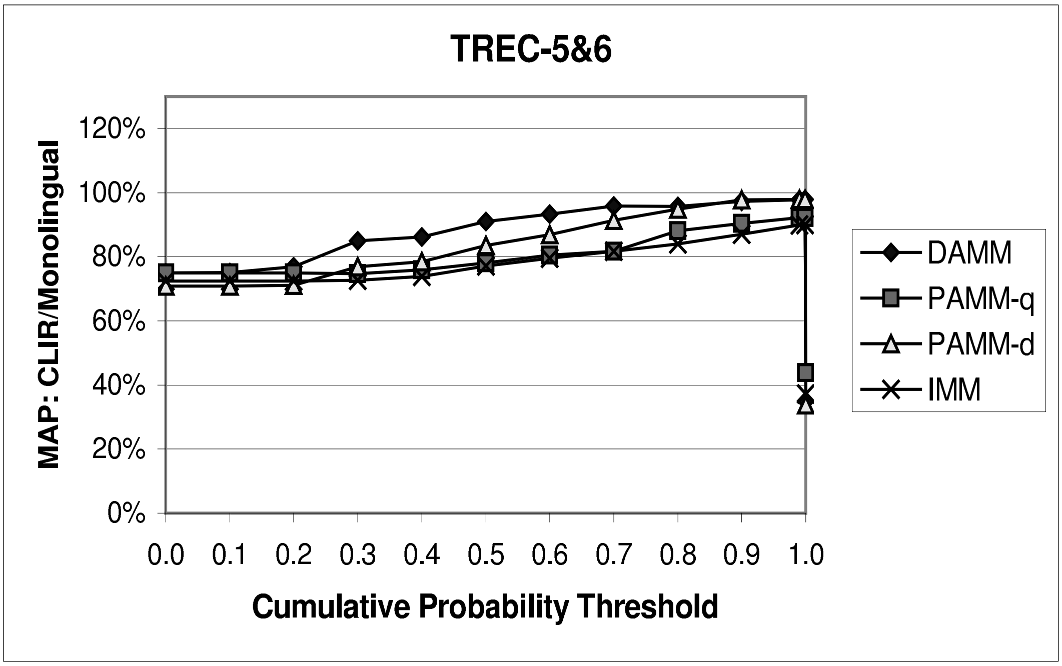


Figure 5.4: Comparison of IMM, PDT, and PSQ (TREC). IMM: Individual Meaning Matching; PSQ: Probabilistic Structured Queries; PDT: Probabilistic Document Translation.



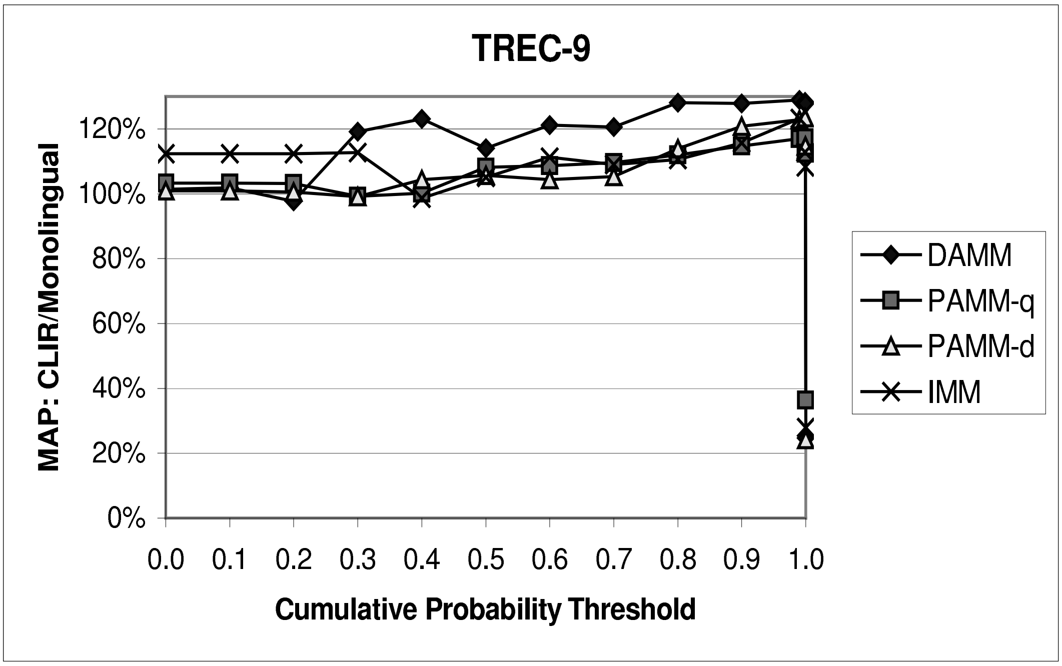


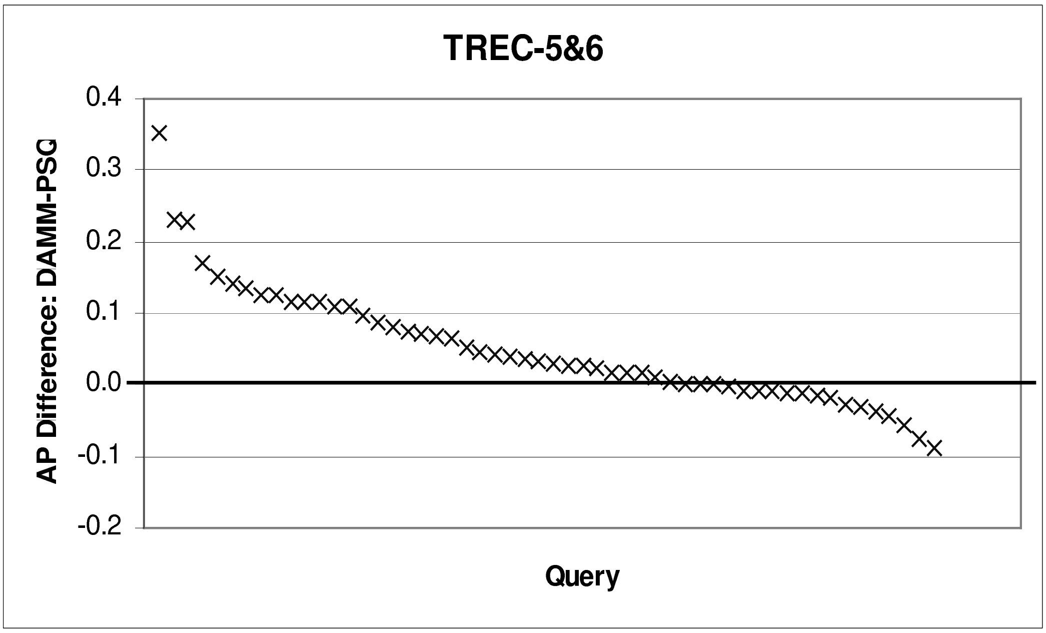
Figure 5.5: Matching meaning with bidirectional translation knowledge. DAMM: Disconnected Aggregated Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; IMM: Individual Meaning Matching.

	PSQ	PDT	APDT	IMM	PAMM-q	PAMM-d	DAMM
MAP (absolute)	0.265	0.284	0.312	0.291	0.298	0.316	0.316
MAP (CLIR/MONO)	82%	88%	97%	90%	92%	98%	98%
$\approx$ MONO?	No	Yes	Yes	No	No	Yes	Yes
Beats PSQ?	N/A	No	Yes	Yes	Yes	Yes	Yes

Table 5.4: Best CLIR mean average precision for different variants of meaning matching. TREC-5&6 collection. 54 English TD queries, 164,789 Chinese documents. Statistical translation models: 10 IBM Model 1 iterations, translations pruned at the cumulative probability threshold of 0.99.

	PSQ	PDT	IMM	PAMM-q	PAMM-d	DAMM
MAP (absolute)	0.284	0.308	0.276	0.287	0.302	0.314
MAP (CLIR/MONO)	116%	126%	113%	117%	124%	128%
$\approx$ MONO?	Yes	Yes	Yes	Yes	Yes	Yes
Beats PSQ?	N/A	No	No	Yes	Yes	Yes

Table 5.5: Best CLIR mean average precision for different variants of meaning matching. TREC-9 collection: 25 English TD queries, 127,758 Chinese documents. Statistical translation models: 10 IBM Model 1 iterations, translations pruned at the cumulative probability threshold of 0.99.



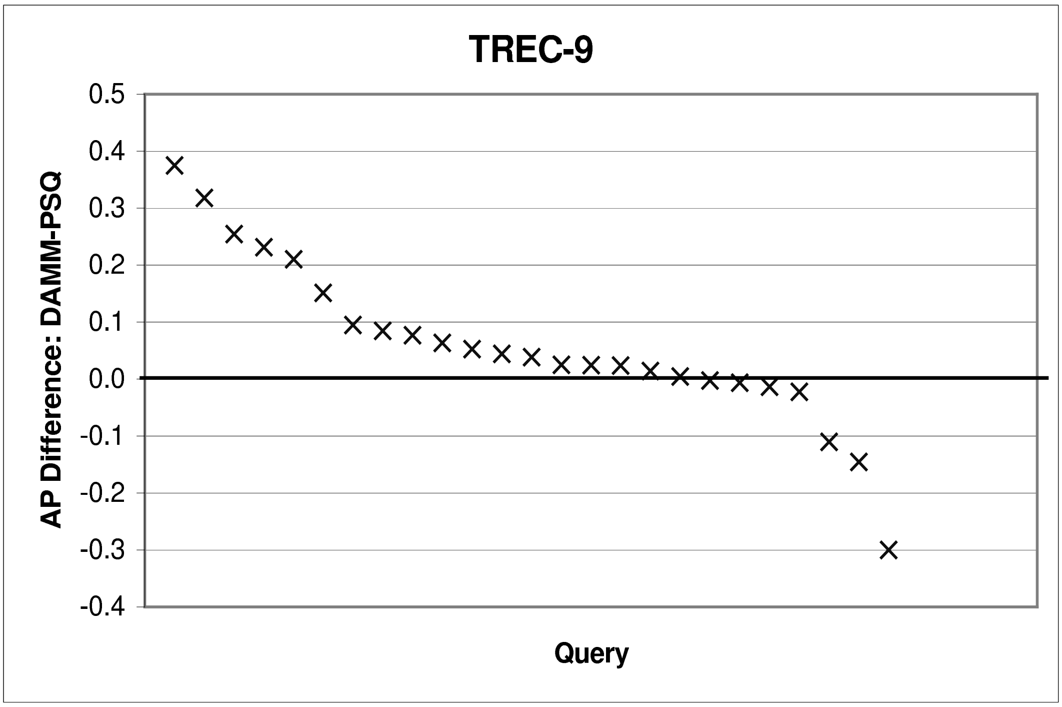


Figure 5.6: Query-by-query comparison of the best DAMM and the best PSQ (TREC). AP: (uninterpolated) average precision DAMM: Disconnected Aggregated Meaning Matching; PSQ: probabilistic structured queries.

CLIR effectiveness over using query translation knowledge alone (PSQ). In addition, each of them was statistically indistinguishable from the monolingual baselines (see Table 5.5 and 5.4). Among all the variants of meaning matching, DAMM achieved the best effectiveness for both TREC-5&6 and TREC-9 (with a MAP of 98% and 128% respectively). These findings for English-Chinese CLIR are consistent with those for English-French CLIR, showing the robustness of meaning matching across different language pairs.

When the best MAP is considered, all four variants of meaning matching that involve bidirectional translation were statistically indistinguishable from each other. However, the best IMM was statistically indistinguishable from the best PSQ for TREC-9 (as shown above) while each of the best of PAMM-q, PAMM-d, and DAMM was significantly better the best PSQ. This demonstrates that both bidirectional translation knowledge and synonymy knowledge can help, and combining them can help more.

To see the proportion of queries that contributed to the observed difference of MAP between DAMM PSQ, we plotted a query-by-query comparison of the best DAMM and the best PSQ for both TREC-5&6 and TREC-9 (see Figure 5.6). Among the 54 TREC-5&6 topics, 39 had higher (uninterpolated) Average Precision (AP) with DAMM, of which 22 achieved an AP difference of at least 0.05. On the other hand, the remaining 15 topics had higher AP with PSQ, of which only 3 had an AP difference of at least 0.05 (the difference never exceeded 0.1). In the case of the 25 TREC-9 topics, 18 topics were better with DAMM, while only 7 topics were better with PSQ. As in the CLEF English-French CLIR experiments,

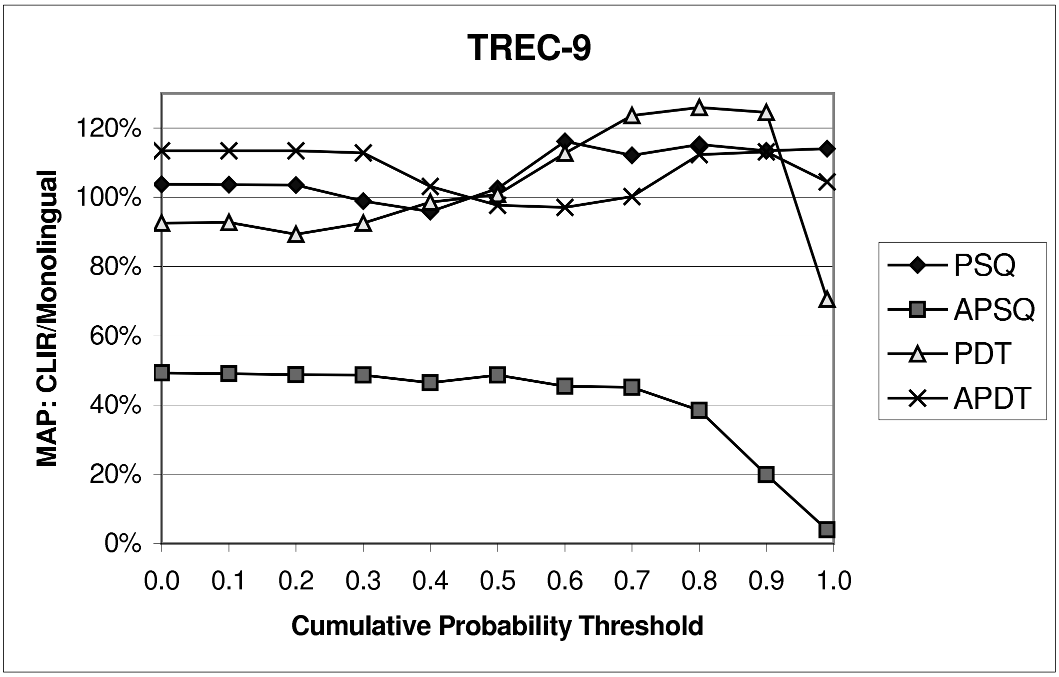


Figure 5.7: The effect of aggregation on unidirectional translation (TREC-9). PSQ: Probabilistic Structured Queries; APSQ: Aggregated Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation.

the analysis here confirms that the observed statistically significant differences in retrieval effectiveness between DAMM and PSQ were not dominated by a small set of topics.

5.2.6 Combining Unidirectional Translation Knowledge with Synonymy Knowledge: APSQ and APDT

Combining unidirectional translation knowledge with statistical synonymy knowledge was performed only in the experiments with TREC-9 collection. As shown in

Figure 5.7, aggregation gave mixed results. For query translation it always significantly degraded, but for document translation it helped in low CPT region while degrading in high CPT region.

We suspect the different aggregation effect on PSQ and PDT was due to the different characteristics of Chinese and English. Many Chinese words often had single-character Chinese as their synonyms in the statistical synonym dictionary derived from statistical translation models. When aggregation was performed on the Chinese translations of English words, some single-character Chinese translations with low-translation probability were grouped into synsets that also contained high-probability translations. The probabilities of those single-character Chinese translations were boosted in this way by aggregation, hence they contributed more to document ranking than when aggregation was not used. Single-character Chinese words could have many different meanings, so many documents not on topic could be retrieved due to the presence of such translations. Word segmentation may worsen the effect by producing incorrect segmentation.

Aggregation on the query side (in English) is quite different. Many statistical synonyms of English words were morphological variants. By grouping them through aggregation, we could boost their translation probabilities, and they could contribute more to document ranking than without aggregation. This could improve recall but not degrade precision. However, the effect was limited to aggregation of synsets with high probabilities, that is, at low CPT thresholds. When we moved to high CPT thresholds, a growing number of low-probability synsets were included. Since low-probability synsets were more likely to contain wrong translations than high-probability synsets, aggregation at high CPT thresholds would not help.

Compared to the results of the CLEF English-French experiments, Aggregation exhibited a consistent pattern of effects. Specifically, aggregation always degraded query translation, while giving mixed effects on document translaltion. However, the effect (both positive and negative) on the English-Chinese CLIR was bigger than on the English-French CLIR. Besides the difference between the two language pairs, the difference between the accuracy of the translation models used in the two experiments may also play an important role.

5.2.7 Using More Accurate Translation Models

Statistical MT training in the above experiments involved only IBM Model 1 iterations. Table 5.6 shows CLIR results with the TREC-9 collection when 5 HMM iterations were added after Model 1 training. APSQ was not tried in this case since previous experiments consistently showed it did not work. Figure 5.8 displays meaning matching using unidirectional translation, and Figure 5.9 displays meaning matching using bidirectional translaltion.

Overall, meaning matching performed quite well, with the best of each variant achieved retrieval effectiveness just as good as the monolingual baseline. There is no statistically significant difference among each pair of them except that the best of DAMM significantly outperformed the baseline PSQ, showing again that combining bidirectional translation knowledge and statistical synonymy knowledge could improve CLIR effectiveness (see Table 5.7).

CDF	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99	0.999	1
PSQ	0.253	0.253	0.251	0.245	0.243	0.287	**0.289**	0.281	0.276	0.274	0.277	N/A	N/A
PDT	0.224	0.224	0.227	0.223	0.223	0.237	0.263	0.292	0.300	**0.306**	0.17	N/A	N/A
APDT	0.246	0.246	0.247	0.247	0.222	0.223	0.202	0.185	0.183	**0.267**	0.241	N/A	N/A
IMM	0.275	0.275	0.275	0.276	0.241	0.257	0.272	0.266	0.270	0.283	**0.301**	0.294	0.1346
PAMM-q	0.252	0.252	0.252	0.243	0.245	0.265	0.266	0.268	0.274	0.281	**0.286**	0.279	0.1141
PAMM-d	0.247	0.247	0.246	0.242	0.255	0.258	0.255	0.258	0.279	0.295	**0.300**	0.294	0.0573
DAMM	0.248	0.249	0.239	0.291	0.301	0.279	0.296	0.295	0.313	0.313	**0.315**	0.315	0.0483

Table 5.6:

CLIR mean average precision based on different variants of meaning matching model. TREC-9 collection: 25 English TD queries, 127,758 Chinese documents. Statistical translation models: 10 IBM Model 1 iterations followed by 5 HMM iterations, cut off at cumulative probability threshold of 0.99. PSQ: Probabilistic Structured Queries; APSQ: Aggregated Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation; IMM: Individual Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in Query language; PAMM-d: Partial Aggregated Meaning Matching in Document language; DAMM: Disconnected Aggregated Meaning Matching

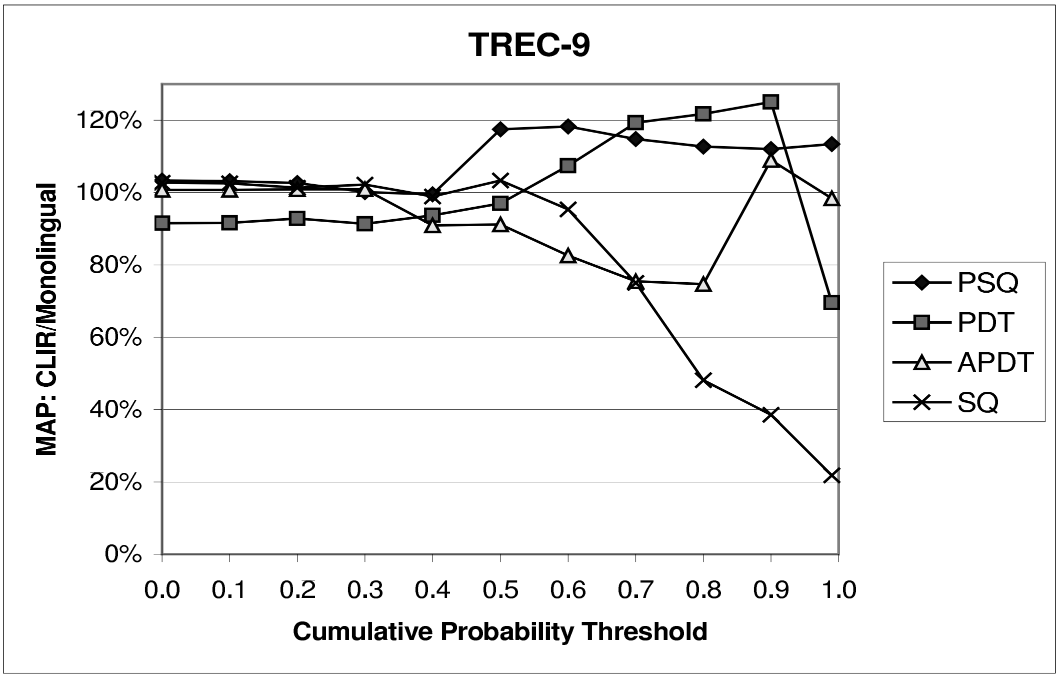


Figure 5.8: Meaning matching using unidirectional translation knowledge (TREC-9, Model 1 + HMM). SQ: Structured Queries; PSQ: Probabilistic Structured Queries; PDT: Probabilistic Document Translation; APDT: Aggregated Probabilistic Document Translation.

	PSQ	PDT	APDT	IMM	PAMM-q	PAMM-d	DAMM
MAP (absolute)	0.289	0.306	0.266	0.301	0.286	0.300	0.315
MAP (CLIR/MONO)	118%	125%	108%	123%	117%	123%	129%
≈ MONO?	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Beats PSQ?	N/A	No	No	No	No	No	Yes

Table 5.7: Comparison of the best MAP of each variant of meaning matching. (TREC-9, MODEL 1 + HMM)

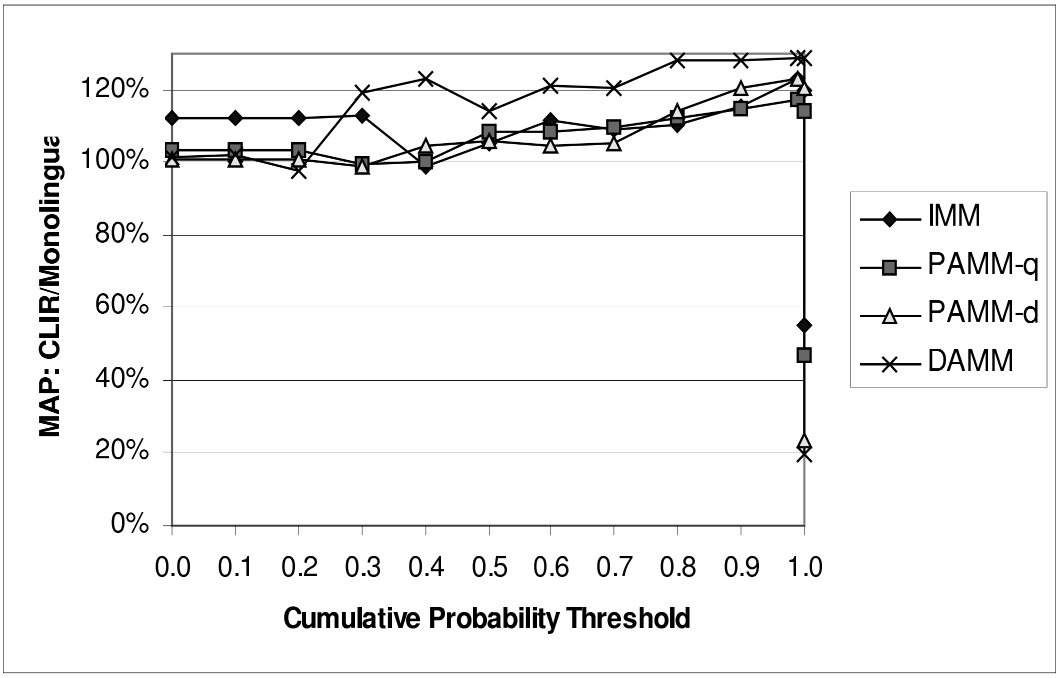


Figure 5.9: Meaning matching using bidirectional translation knowledge (TREC-9, Model 1 + HMM). IMM: Individual Meaning Matching; PAMM-q: Partial Aggregated Meaning Matching in query language; PAMM-d: Partial Aggregated Meaning Matching in document language; DAMM: Disconnected Aggregated Meaning Matching.

Run	MSRCN	FDU	BBN	CHUHK	PIRCS	UMass	IBM	DAMM
MAP	0.308	0.291	0.287	0.258	0.245	0.242	0.240	0.314

Table 5.8: Comparison with the Best Reported TREC-9 CLIR Runs.

Adding HMM iteration in statistical MT training did not lead to significant improvement of CLIR effectiveness. For example, the best meaning matching (DAMM) achieved about the same MAP with or without HMM iterations during statistical MT training. This suggests that it may be unnecessary, when meaning matching is used for CLIR, to train statistical translation models with more complicated and inefficient IBM Models or HMM iterations. However, In order to get a full understanding, future studies are needed for the influence of different parameter settings in statistical MT training on the effectiveness of meaning matching.

5.2.8 Comparison with the Best Official Runs

Although it is close to the median of the official runs, our monolingual baseline for TREC-9 is lower than the best of them. Achieving CLIR effectiveness comparable to this monolingual baseline does not necessarily justify that the CLIR technique involved is effective. For this reason, we want to compare our CLIR results with the top official CLIR runs that used the same test collection. Table 5.8 shows the top 7 TREC-9 CLIR runs and the best run in our study (DAMM). Among these runs, the best (MSRCN) used pre-translation expansion and complex translation disambiguation techniques [22]. The second best (FDU) used synonymy resources and phrase translation [84]. The third best (BBN) used language modeling approach and did not involve expansion [85]. Our best run (DAMM), which did not use phrase translation or query expansion, still achieved a higher MAP than any official TREC-9 run. However, due to lack of detailed results of these runs, statistical significance tests were not conducted. We should point out that those official runs were performed under time constraint and people's knowledge of the test collection at that time was limited.

5.3 Summary

In this chapter, we presented our English-Chinese CLIR experiment results that involved one test collection from TREC-5 and TREC-6, and another from TREC-9. In both cases, we showed that cross-langauge meaning matching that combined bidirectional translation knowledge and statistical synonymy knowledge significantly outperformed the probabilistic structured query technique, which used translation knowledge from query language to document language alone. The effectiveness of the technique is comparable to monolingual baselines that we gained by searching Chinese documents with human-translated Chinese queries. Equally importantly, it is at least as good as the best reported CLIR effectiveness, which used additional techniques such as blind relevance feedback.

These findings are consistent with what we observed from the English-French CLIR experiments, showing the robustness of the meaning matching model across different language pairs, test collections, and statistical translation accuracy. However, probabilistic document translation exhibited some difference as compared to the English-French CLIR experiment. Specifically, its effectiveness decreased more sharply as more low-probability were used in the English-Chinese experiments than in the English-French experiments. We suspect it is because the larger difference in vocabulary size between English and Chinese than between English and French.

English-Chinese CLIR presented the important issue of word segmentation. It directly affected word-based retrieval as wrong segmentations could lead to the retrieval of non-relevant documents. Our failure analysis of the monolingual baseline run with the TREC-9 collection confirmed the effect of wrong segmentation. In addition, we found that some inappropriate human translations also account for the decrease of monolingual retrieval effectiveness. As a result, all variants of the meaning matching model achieved CLIR effectiveness comparable to monolingual performance in the best case. We believe that word segmentation in Chinese also had an effect on the accuracy of the statistical translation models, and our analysis of the difference between aggregation in the two languages seems to support the speculation.

Finally, within our framework of cross-language meaning matching, adding HMM iterations in training statistical translation models did not lead to the improvement of CLIR effectiveness over using IBM Model 1 iterations alone. This suggests that it may be unnecessary to use more complex and inefficient IBM Models and in training statistical translation models for the purpose of CLIR.

Chapter 6

Conclusions

The most important contribution of the study is the introduction of the meaning matching model for cross-language information retrieval, based on the notion that information retrieval fundamentally depends upon matching what the searcher means with what the document author meant. The model defines the probability two terms share the same meaning. For any pair of terms, their meaning matching probability can be computed by combining translation knowledge in both directions and synonymy knowledge in both languages. The probability then can be used to estimate the term frequency and document frequency of each query term. We showed the probabilistic structured query method is a special case of our meaning matching model when only query translation knowledge is available. We also introduced probabilistic document translation, another special case in which only document translation knowledge is available.

The meaning model is relatively simple and intuitive. The model can easily achieve CLIR effectiveness that is at least as good as the best results reported in previous studies that used the same test collections but more complex techniques, such as translation disambiguation. In addition, the model can accommodate translation uncertainty to a great extent. For example, in all the experiments in which bidirectional translation knowledge was used, applying a cumulative probability threshold of 0.9 to pruning synonyms always led to CLIR effectiveness that was better than one-best selection.

Through our experiments with an English-French test collection and two English-Chinese test collections, we were able to answer five research questions listed at the beginning of the dissertation:

1. *When using bidirectional translation knowledge and synonymy knowledge, can CLIR based on meaning matching significantly outperform the probabilistic structured query method, which uses only query translation knowledge?* All the three experiments showed that when bidirectional translation knowledge and statistical synonymy knowledge were used, the best case of cross-language meaning matching significantly outperformed the best case of probabilistic structured queries.

2. *When using translation knowledge from document language to query language alone, can CLIR based on meaning matching achieve retrieval effectiveness comparable to the probabilistic structured query method?* The probabilistic document retrieval technique introduced in the study, which uses only document translation knowledge, performed at least as well as the probabilistic structured query method in all our experiments. In fact, the best case of the probabilistic document translation always achieved higher mean average pre-

cision than the best case of the probabilistic structured queries, although the difference was not statistically significant.

3. *How can we establish a fair monolingual baseline to which CLIR effectiveness can be compared?* In our study, we used statistical translation models in two directions to identify synonyms, which were then used for query expansion. Our English-French CLIR experiments showed that query expansion with statistical synonyms acquired in this way could lead to significant improvement of monolingual retrieval effectiveness when stemming was not used. However, when stemming was applied, query expansion with statistical synonyms had little effect.

4. *When using bidirectional translation knowledge and synonymy knowledge, can CLIR based on meaning matching achieve effectiveness comparable to monolingual effectiveness?* In all experiments, the best case of cross-language meaning matching that used bidirectional translation knowledge and synonymy knowledge achieved retrieval effectiveness comparable to monolingual effectiveness under the same experiment condition, with relative mean average precision of at least 98%. Among the three monolingual baselines, two (CLEF and TREC-5&6) were comparable to the best reported results gained from experiments with the same test collections. The third one (TREC-9) was equal to the median of the reported official runs, but CLIR effectiveness we gained was at least as good as the best reported result.

5. *How does the effectiveness of the meaning matching model change according*

to the number of translation alternatives used, and how can translation alternatives be pruned to maximize CLIR effectiveness? CLIR based on meaning matching that uses bidirectional translation knowledge and synonymy knowledge was robust to the inclusion of spurious translations. In all experiments, cross-language meaning matching remained as effective as monolingual retrieval even when the cumulative probability reached 0.9. However, when all translation alternatives were used, the effectiveness could degrade dramatically. Therefore, it is important to prune translations. Our study showed using the cumulative probability threshold could help choose translation alternatives, so CLIR achieved effectiveness comparable to its monolingual counterpart. The technique first ranks translation alternatives in decreasing order of their probabilities, then selects translations from the top until their cumulative probability reaches a certain value. Furthermore, we showed it is useful to prune statistical translation models before they are used in the derivation of the meaning matching model.

6.1 Limitations

Despite the development of the meaning matching model and the significant improvement of CLIR effectiveness due to the model, the study has obvious limitations. Some are common to all IR experiments that involve the use of test collections, while others are unique to the study.

The meaning matching model naturally pointed us to use synsets aligned across languages, and we tested the idea with EuroWordNet. Unfortunately, the results showed full aggregated meaning matching based on EuroWordNet was significantly worse than other variants of meaning matching. The main reason is many high-probability translations (usually accurate translations) are not covered by EuroWordNet, hence were missed by the full aggregated meaning matching model. A better way to handle the situation in which there is not a pair of aligned synsets connecting to words might be to back off to some other variant of meaning matching. This way, we may be able to recover lots of high-probability translations missed by full aggregated meaning matching.

The way we selected statistical synonyms in our study may be oversimplified. As we have shown, each of the synonyms derived from statistical translation models has a probability that specifies the likelihood it sharing the same meaning with the original term. The probability is useful for the selection of synonyms in monolingual query expansion and in grouping synonymous translations into synsets. The selection of synonyms will have an important influence on the retrieval effectiveness. In our cross-language meaning matching experiments, however, we simply excluded synonyms whose probability was smaller than 0.1. In better selection solution, we could use a cumulative probability threshold, or at least try several individual probability thresholds so that we can see how the effectiveness of the meaning matching model changes accordingly.

We described two methods of statistically conflating synonymous translations to represent "meaning." The greedy method assigns a translation to only the most probable synset, and the conservative method assigns a translation to all its synsets with uniform probability. In our study, we tried only the greedy method as we suspect it is better. It may be interesting to test the conservative method. Furthermore, we might consider other ways to group synonymous translations into synsets. For example, orthographic similarity could be a useful cue for deciding which translations should belong to the same synsets. The conservative methods that assigns a translation to all its synsets with equal probability might be improved by using unequal probability, proportionally depending on the total unigram frequencies of the translations each synset contains. Also, the statistical synonymy knowledge could be enhanced by merging with WordNet synsets. One way to do that would be to assign terms in each WordNet synset with uniform probability, then linearly combining them with statistical synonyms.

A practical limitation of our present implementation of the meaning matching model lies in its efficiency. In a CLIR system for end users, retrieval effectiveness is not the only factor that influences their satisfaction with the system. Other aspects, such as the system's response time to their information requests, are also important. For example, in our English-French CLIR experiments, on average it took 6 seconds to process a query for the best case of meaning matching, but only about 0.1 second to process a query with one-best translation. However, this issue can be partially addressed by computing weights at indexing time (in our experiments, we used a query-time implementation).

Still another limitation of the meaning matching model is that it relies on sentence-aligned parallel texts, which may not be readily available for some language pairs. Moreover, our study tested only the effectiveness of the meaning matching model with statistical translations obtained from clean parallel corpus. It is unclear how it will perform if translation knowledge is obtained from noisier sources, such as automatically aligned parallel Web pages.

In our English-Chinese CLIR experiments, words were used as index terms. However, BBN reported combining Chinese character bigrams and character unigrams outperformed using words alone [85]. However, additional issues such as post-translation resegmentation arise when translation and indexing units are not matched [63], and it is unclear how meaning matching can or should be integrated with post-translation resegmentation.

Finally, actual users were never involved, though the ultimate goal of the study is to design models and techniques to help them find information. As in all other IR experiments that use test collections, the user's information needs in our study were simply represented as a fixed sets of search topics. In reality, information needs may change as users interact with the system and the retrieved documents. Therefore, our study addressed only a small and static part of a large and dynamic process of information seeking. Also, it is not clear whether the cross-language meaning matching model actually retrieved relevant documents more useful than those retrieved with probabilistic structured queries. In other words, a quantitative effectiveness measure such as mean average precision is not sufficient for a full understanding of a model's utility. A study that involves users examining retrieved documents, for example, might provide insight into the utility of additional documents.

6.2 Future work

The meaning matching model may be tested in other situations not covered in this study. A potential area of improvement for the model may lie in the use of phrase translation. Many CLIR studies have revealed phrase translation can reduce translation ambiguity. Recent studies on alignment templates in statistical MT have also showed translation based on learned phrases can be more accurate than translation based solely on individual words [66], directing us toward integrating phrase translation into our cross-language meaning matching model. Perhaps, we should integrate word sense disambiguation into the meaning matching model. There has been intensive research in the field of word sense disambiguation [19]. We expect some shallow word sense disambiguation will exclude obviously incorrect or impossible translations produced by a statistical translation model, which will make the meaning matching model more robust to the remaining translations. In addition, since clean sentence-aligned corpora are rare for many languages, exploring the usefulness of resources such as parallel Web pages for the meaning matching model might increase the number of the language pairs that it can be used for. Techniques suggested by Resnik indicate it is a promising direction [72].

We tested only the effectiveness of the meaning matching model in the framework of Okapi BM25 weights. The estimated meaning matching probability between terms, however, can be employed in any IR model that uses term frequency and/or document frequency in computing term weight. It might be interesting to integrate the meaning matching model into a vector space model, or to substitute it for the document translation model used in retrieval based on language models.

The idea of using bidirectional translation knowledge and synonymy knowledge could also be useful for interactive cross-language information retrieval. It can lead to better ranked lists of documents and it can help searchers in making better translation selection decision. In previous studies, query language terms that share the same translation as a selected query term were displayed to the searcher to help define the meaning of a proposed translation (for example, [28]). Synsets could be used as a basis for pruning the displayed "definition," perhaps leading to better decisions by the searcher. Statistical synonyms identified using bidirectional translation may be directly used for interactive monolingual query refinement as well. Searchers could browse the list of synonym sets (perhaps together with synset probabilities) for each query term to decided, selecting entire synonym sets when appropriate, and perhaps deleting individual "synonyms" in those sets when necessary. This way, searchers would have a richer pool of terms to explore when refining initial queries.

The meaning matching model could also be extended to other information retrieval tasks that require search under uncertainty. CLIR by nature involves uncertainty - given a query term, we are not certain which terms in each document share the same meaning. The meaning matching model attempts to address these issues and investigate techniques that can use meaning matching probability to improve CLIR effectiveness. The improvement achieved in our study allows us to believe it is a reasonable way to deal with translation uncertainty. Other IR tasks share similar characteristics. For example, in speech retrieval, documents are converted from audio into text, which is analogous to document translation in CLIR. In this case, the probabilistic document translation technique introduced in this study may be useful. Similar idea could also be applied to document image retrieval, in which document images are usually converted into text with optical character recognition techniques. Again, the probabilistic document retrieval method may help recover correct strings otherwise missed by one-best recognition methods.

Finally, we were only able to investigate the effectiveness of the meaning matching model with two language pairs. Human languages are complex, and each language possesses unique features that might affect retrieval. We anticipate using the meaning matching model in cross-language information retrieval with other language pairs.

6.3 Summary

The use of translation probabilities has become one of the most promising areas in cross-language information retrieval for the last several years. This dissertation represents a major research effort to advance the state-of-the-art in integrating statistical translation knowledge and synonymy knowledge into a unified framework.

Ranking documents based on the TF and DF of query terms that they contain has been one of the most commonly used approaches to information retrieval, as it has worked well in monolingual applications. Meaning matching extends this in a natural way to the CLIR case. Specifically, the TF and DF of query terms are estimated separately across languages by integrating the probabilities that query terms and document terms sharing the same meaning. Our experiment results indicate that meaning matching works as effectively as the best known CLIR techniques, and nearly as well as monolingual techniques. This provides a rich basis for future work on CLIR and other types of information retrieval under uncertainty.

As we are building a global information infrastructure, it has become not only desirable but also necessary to share information across language boundaries. The explosive growth of electronic information in many languages demands techniques that can help people to effectively access such information. The work reported in this dissertation provides a unique perspective on how the task of searching information across languages can be accomplished.

BIBLIOGRAPHY

[1] Lisa Ballesteros. Cross language retrieval via transitive translation. In W. Bruce Croft, editor, *Advances in Information Retrieval: Recent Research from the CIIR*, pages 203–234. Kulwer Academic Publishers, 2000.

[2] Lisa Ballesteros and W. Bruce Croft. Phrasal translation and query expansion techniques for cross-language information retrieval. In *Proceedings of the 20th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 84–91. ACM Press, July 1997.

[3] Lisa Ballesteros and W. Bruce Croft. Resolving ambiguity for cross-language retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 64–71. ACM Press, August 1998.

[4] Adam Berger and John Lafferty. Information retrieval as statistical translation. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–229. ACM Press, August 1999.

[5] M. Boughanem, C. Chrisment, and N. Nassr. Investigation on disambiguation in CLIR: Aligned corpus and bi-directional translation-based strategies. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum*, pages 158–167. Springer-Verlag GmbH, 2001.

[6] P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[7] Chris Buckley, Mandar Mitra, Janet Walz, and Claire Cardie. Using clustering and SuperConcepts within SMART: TREC 6. In *The Six Text REtrieval Conference (TREC-6)*, 1997. http://trec.nist.gov/.

[8] Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic query expansion using smart: TREC 3. In *The Third Text REtrieval Conference (TREC-3)*, 1994. http://trec.nist.gov/.

[9] Chris Buckley and Ellen E. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 33–40. ACM Press, 2002.

[10] Aitao Chen. Cross-language retrieval experiments at CLEF-2002. In Carol Peters, Martin Braschler, Julio Gonzalo, and Michael Kluck, editors, *Advances in Cross-Language Information Retrieval: Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002.*

[11] Kenneth W. Church and Patrick Hanks. Word association norms, mutual information and lexicography. In *Proceedings of the 27th Annual Conference of the Association of Computational Linguistics*, pages 76–82, 1989.

[12] C. W. Cleverdon. Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems. Report, College of Aeronautics, Cranfield England, 1962.

[13] C. W. Cleverdon, J. Mills, and E. M. Keen. Factors determining the performance of indexing systems, vol.1 : Design, vol.2: Test results. Aslib cranfield research project, Cranfield England, 1966.

[14] W. Bruce Croft. Knowledge-based and statistical approaches to text retrieval. *IEEE Expert*, 8(2):8–12, 1993.

[15] Kareem Darwish and Douglas W. Oard. Probabilistic structured query methods. In *Proceedings of the 21st Annual 26th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 338–344. ACM Press, July 2003.

[16] Mark Davis and Ted Dunning. A TREC evaluation of query translation methods for multilingual text retrieval. In *The Fourth Text Retrieval Conference (TREC-4)*. National Institute of Standards and Technology, November 1995. http://trec.nist.gov/.

[17] Mark Davis and William C. Ogden. QUILT: Implementing a large-scale cross-language text retrieval system. In *Proceedings of the 20th International ACM*

SIGIR Conference on Research and Development in Information Retrieval, pages 92–98. ACM Press, July 1997.

[18] 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*, pages 175–185. Evolutionary Programming Society, March.

[19] Mona T. Diab. *Word Sense Disambiguation within a Multilingual Framework.* Ph.D. thesis, University of Maryland, 2004.

[20] Dina Demner Fushman and Douglas W. Oard. The Effect of Bilingual Term List Size on Dictionary-Based Cross-Language Information Retrieval. Technical Report LAMP-TR-097,CS-TR-4452,UMIACS-TR-2003-22, University of Maryland, College Park, February 2003.

[21] Bonnie J. Dorr, Daqing He, Jun Luo, Douglas W. Oard, Richard Schwartz, Jianqiang Wang, and David Zajic. iCLEF 2003 at Maryland: Translation selection and document selection. In *Comparative Evaluation of Multilingual Information Access Systems: 4th Workshop of the Cross-Language Evaluation Forum*, pages 435–449. Springer-Verlag GmbH, 2003.

[22] Jianfeng Gao, Jian-Yun Nie, Jian Zhang, Endong Xun, Yi Su, Ming Zhou, and Changning Huang. TREC-9 CLIR experiments at MSRCN. In *The Nineth Text REtrieval Conference (TREC-9)*. National Institute of Standards and Technology, November 2000. http://trec.nist.gov/.

[23] Fredric C. Gey, Hailing Jiang, Aitao Chen, and Ray R. Larson. Manual queries and machine translation in cross-language retrieval and interactive retrieval with Cheshire II at TREC-7. In *The Seventh Text REtrieval Conference*, pages 527–540. National Institutes of Standards and Technology, November 1998. http://trec.nist.gov.

[24] Tim Gollins and Mark Sanderson. Improving cross language information retrieval with triangulated translation. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 90–95. ACM Press, 2001.

[25] Donna Harman. Towards interactive query expansion. In *Proceedings of the 11th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 321–331. ACM Press, 1988.

[26] Donna Harman. Overview of the first TREC. In *The First Text REtrieval Conference*. National Institutes of Standards and Technology, November 1992. http://trec.nist.gov.

[27] Daqing He, Douglas W. Oard, Jianqiang Wang, Jun Luo, Dina Demner-Fushman, Kareem Darwish, Pilip Resnik, Sanjeev Khudanpur, Michael Nossal, and Anton Leuski. Making MIRACLEs: Interactive translingual search for Cebuano and Hindi. *ACM Transaction on Asian Language Information Processing*, 2(3):219–244, 2003.

[28] Daqing He, Jianqiang Wang, Douglas W. Oard, and Michael Nossal. Comparing user-assisted and automatic query translation. In *Advances in Cross-Language Information Retrieval: Third Workshop of the Cross-Language Evaluation Forum*, pages 400–415. Springer-Verlag GmbH, 2002.

[29] Djoerd Hiemstra. A linguistically motivated probabilistic model of information retrieval. In *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, pages 569–584. Springer, 1998.

[30] Djoerd Hiemstra. *Using Language Models for Information Retrieval.* Ph.D. thesis, University of Twente, 2000.

[31] David Hull. Using statistical testing in the evaluation of retrieval experiments. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 329–338. ACM Press, 1993.

[32] David A. Hull. Using structured queries for disambiguation in cross-language information retrieval. In *AAAI Symposium on Cross-Language Text and Speech Retrieval*, pages 84–98. American Association for Artificial Intelligence, March 1997.

[33] Karen Sparck Jones. Automatic indexing. *Journal of Documentation*, 30:393–432, 1974.

[34] Karen Sparck Jones. *Information Retrieval Experiment.* Butterworths, London, 1981.

[35] Karen Sparck Jones and C. J. Van Rijsbergen. Report on the need for and provision of an "ideal" information retrieval collection. In *British Library Research and Development Report 5266*, Computer Laboratory, University of Cambridge, 1975.

[36] In-Su Kang, Seung-Hoon Na, and Jong-Hyeok Lee. POSTECH at NTCIR-4: CJKE monolingual and Korean-related cross-language retrieval experiments. In *Working Notes of the 4th NTCIR Workshop.* National Institute of Informatics, 2004. http://research.nii.ac.jp/ntcir/index–en.html.

[37] Kevin Knight and Johnathan Graehl. Machine transliteration. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 128–135, 1997.

[38] Philipp Koehn. Europarl: A multilingual corpus for evaluation of machine translation. unpublished draft. 2002.

[39] Wessel Kraaij. *Variations on Language Modeling on Information Retrieval.* Ph.D. thesis, University of Twente, 2004.

[40] John Lafferty and Chengxiang Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 111–119. ACM Press, 2001.

[41] Leah S. Larkey and Margaret E. Connel. Automatic information retrieval at UMass in TREC-10. In *The Tenth Text REtrieval Conference*. National Institutes of Standards and Technology, November 2001. http://trec.nist.gov.

[42] Victor Lavrenko, Martin Choquette, and W. Bruce Croft. Corss-lingual relevance models. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 175–182. ACM Press, August 2002.

[43] Victor Lavrenko and W. Bruce Croft. Relevance-based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 120–127. ACM Press, 2001.

[44] T. Leek, H. Jin, S. Sista, and R. Schwartz. The BBN cross-lingual topic detection and tracking system. In *Working Notes of the Third Topic Detection and Tracking Workshop*. National Institutes of Standards and Technology, February 2003.

[45] Gina-Anne Levow and Douglas W. Oard. Evaluating lexical coverage for cross-language information retrieval. In *Workshop on Multilingual Information Processing and Asian Language Processing*, pages 69–74, February 2000.

[46] Gina-Anne Levow and Douglas W. Oard. Translingual topic tracking with PRISE. In *Working Notes of the Third Topic Detection and Tracking Workshop*. National Institutes of Standards and Technology, February 2000.

[47] Chuna-Jie Lin, Wen-Cheng Lin, Guo-Wei Bian, and Hsin-Hsi Chen. Description of the NTU japanese-english cross-lingual information retrieval system. In *Proceedings of the first NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*. National Institute of Informatics, September 1999. http://research.nii.ac.jp/ntcir/index–en.html.

[48] M. Littman, S. Dumias, and T. Landauer. Automatic cross-language information retrieval using latent semantic indexing. In G. Grefenstette, editor, *Cross-Language Information Retrieval*. Kluwer, 1998.

[49] H. Ma, B. Karagol-Ayan, D. Doermann, D. Oard, and J. Wang. Parsing and tagging of bilingual dictionaries. *Traitement Automatique Des Langues*, 44(2):125–150, 2003.

[50] James Mayfield, Paul McNamee, and Christine Piatko. The JHU/APL HAIRCUT system at TREC-8. In *The Eighth Text REtrieval Conference (TREC-8)*. National Institutes of Standards and Technology, 1999. http://trec.nist.gov/.

[51] J. Scott McCarley. Should we translate the documents or the queries in cross-language information retrieval? In *Proceedings of the 37th Annual Conference of the Association for Computational Linguistics*, pages 208–214, 1999.

[52] P. McNamee and J. Mayfield. Comparing cross-language query expansion techniques by degrading translation resources. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 159–166. ACM Press, August 2002.

[53] Helen Meng, Berlin Chen, Erika Grams, Sanjeev Khudanpur, Wai kit Lo, Gina Anne-Levow, Douglas Oard, Patrick Schone, Karen Tang, Hsin-Min Wang, and Jianqiang Wang. Mandarin english information (MEI): Investigating translingual speech retrieval. Technical report, Center for Language and Speech Processing, Johns Hopkins University, Oct 2000.

[54] David R. H. Miller, Tim Leek, and Richard M. Schwartz. A Hidden Markov Model information retrieval system. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 214–220. ACM Press, August 1999.

[55] George A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, Nov. 1995.

[56] Christof Monz and Maarten de Rijke. Shallow morphological analysis in monolingual information retrieval for dutch, german, and italian. In *Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum, CLEF 2001*, pages 262–277. Springer-Verlag GmbH, 2001.

[57] Jian-Yun Nie and Michel Simard. Using statistical models for bilingual IR. In *Proceedings of Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum, CLEF 2001*, 2001.

[58] Jian-Yun Nie, Michel Simard, Pierre Isabelle, and Richard Durand. Cross-language information retrieval based on parallel texts and automatic mining

of parallel texts from the Web. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 74–81. ACM Press, August 1999.

[59] Douglas W. Oard and Julio Gonzalo Arroyo. The cross-language evaluation forum (CLEF) 2001 interactive track. In *Presentation at the Cross-Language Evaluation Forum (CLEF) 2001 Workshop*, September 2001.

[60] Douglas W. Oard and Anne R. Diekema. Cross-language information retrieval. In *Annual Review of Information Science and Technology*, volume 33, pages 223–256. American Society for Information Science, 1998.

[61] Douglas W. Oard, Gina-Anne Levow, and Clara I. Cabezas. CLEF experiments at Maryland: Statistical stemming and backoff translation. In *Proceedings of Evaluation of Cross-Language Information Retrieval Systems: Third Workshop of the Cross-Language Evaluation Forum, CLEF 2000*, 2000.

[62] Douglas W. Oard and Jianqiang Wang. Effects of term segmentation on Chinese/English cross-language information retrieval. In *Proceedings of the Symposium on String Processing and Information Retrieval*, pages 149–157, September 1999. http://www.glue.umd.edu/∼oard/research.html.

[63] Douglas W. Oard and Jianqiang Wang. NTCIR-2 ECIR experiments at Maryland: Comparing Pirkola's structured queries and balanced translation. In *Proceedings of the Second NTCIR Workshop on Research in Chinese & Japanese*

Text Retrieval and Text Summarization. National Institute of Informatics, March 2001. http://research.nii.ac.jp/ntcir/index–en.html.

[64] F. J. Och and H. Ney. Improved statistical alignment models. In *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics*, pages 440–447, October 2000.

[65] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[66] Franz Josef Och and Hermann Ney. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4), 2004.

[67] Eugenio Picchi and Carol Peters. Cross language information retrieval: A system for comparable corpus querying. In Gregory Grefenstette, Alan Smeaton, and Páraic Sheridan, editors, *Workshop on Cross-Linguistic Information Retrieval*, pages 24–33. ACM SIGIR, August 1996.

[68] Ari Pirkola. The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 55–63. ACM Press, August 1998.

[69] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281. ACM Press, August 1998.

[70] Martin F. Porter. An algorithm for suffix stripping. *Program*, 14:130–137, 1980.

[71] Yan Qu, Gregory Grefenstette, and David A. Evans. Automatic transliteration for japanese-to-english text retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 353–360. ACM Press, July 2003.

[72] Philip Resnik. Parallel strands: A preliminary investigation into mining the web for bilingual text. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas on Machine Translation and the Information Soup*, pages 72–82. Springer-Verlag London, 1998.

[73] S. E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146, 1976.

[74] S. E. Robertson and Karen Sparck-Jones. Simple proven approaches to text retrieval. Cambridge University Computer Laboratory, 1997.

[75] G. Salton. *The SMART Retrieval System*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.

[76] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[77] Mark Sanderson and Justin Zobel. Information retrieval system evaluation: Effort, sensitivity and reliability. In *Proceedings of the 28th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169. ACM Press, August 2005.

[78] Jacques Savoy. Report on CLEF-2001 experiments: Effective combined query-translation approach. In *Evaluation of Cross-Language Information Retrieval Systems : Second Workshop of the Cross-Language Evaluation Forum*. Springer-Verlag GmbH, 2001.

[79] C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. The University of Illinois Press, Urbana, 1949.

[80] Páraic Sheridan, Jean Paul Ballerini, and Peter Schäuble. Building a large multilingual test collection from comparable news documents. In *Cross-Language Information Retrieval*, pages 56–65. Kluwer Academic, 1998.

[81] Amit Singhal, John Choi, Donald Hindle, and Fernado Pereira. ATT at TREC-7. In *The Seventh Text REtrieval Conference*, pages 239–252. National Institutes of Standards and Technology, November 1998. http://trec.nist.gov.

[82] Robert S. Taylor. Question-negotiation and information seeking in libraries. *College and Research Libraries*, pages 178–194, 1968.

[83] Ellen M. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. *Information Processing & Management*, 36(5):697–716, 2000.

[84] Lide Wu, Xuan jing Huang, Yikun Guo, Bingwei Liu, and Yuejie Zhang. FDU at TREC-9: CLIR, filtering and QA tasks. In *The Nineth Text REtrieval Conference (TREC-9)*. National Institutes of Standards and Technology, November 2000. http://trec.nist.gov/.

[85] Jinxi Xu and Ralph Weischedel. TREC-9 cross-lingual retrieval at BBN. In *The Nineth Text REtrieval Conference*. National Institutes of Standards and Technology, November 2000. http://trec.nist.gov.

[86] Ying Zhang and Phil Vines. Using the web for automated translation extraction in cross-language information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169. ACM Press, 2004.

[87] Justin Zobel. How reliable are the results of large-scale information retrieval experiments? In C.J. Van Rijsbergen W. Bruce Croft, Alistair Moffat, editor, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 307–314. ACM Press, August 1998.