Exploiting Common Search Interests across Languages for Web Search

GAO, Wei

A Thesis Submitted in Partial Fulfilment

of the Requirements for the Degree of

Doctor of Philosophy

in

Systems Engineering and Engineering Management

The Chinese University of Hong Kong

April 2010

UMI Number: 3446025

Thesis/Assessment Committee

Professor WONG Kam-Fai (Chair)

Professsor WONG Kam-Fai (Thesis Supervisor)

Professor LAM Wai (Committee Member)

Professor MENG Mei-Ling (Committee Member)

Professor LEE Dik Lun (External Examiner)

To my family.

ACKNOWLEDGMENTS

The success of this dissertation is attributed to the assistances of many people. First, I thank my academic supervisor, Kam-Fai Wong, for both his financial support and his provisions of freedom on choosing the research goal for me to pursue. Second, special gratitude goes to Dr. Cheng Niu, my mentor at Microsoft Research Asia (MSRA), for his important guidance on this work during my visits to his group. He was always ready for discussions, and most importantly, I learned from him indispensable research skills, such as critical thinking and writing. His advice effectively kept me running towards the rewarding direction. Third, I should pay back with thankfulness to my friend and co-author, Dr. John Blitzer, who was a visiting scholar in MSRA and now a research fellow with UC Berkeley, for his interests and active participation in one of the important parts of this exciting topic. John's excellent expertise on machine learning in NLP provided crucial help to the success of our joint work. Fourth, I also thank Dr. Ming Zhou for his high-level guidance to this study and Prof. Jian-Yun Nie for polishing our papers for publication.

I would feel guilty if not stressing the support from my family. During this study, my wife and my parents are most supportive. They understood my difficulty and constantly reminded me to forget the difficulties of theirs. I present the stories of this study here in a systematic way and bestow this dissertation to my family.

This work was partially done when I was visiting the Natural Language Computing group at MSRA. I am grateful to Dr. Ming Zhou, the group leader, for offering me the valuable opportunities. This work was also partially supported by CUHK direct grant (no.: 2050443) and Microsoft-CUHK Joint Laboratory for Human-Centric Computing and Interface Technologies (under ITF project no.: ITS/182/08).

ABSTRACT

EXPLOITING COMMON SEARCH INTERESTS ACROSS LANGUAGES FOR WEB SEARCH

This work studies something new in Web search to cater for users' cross-lingual information needs by using the common search interests found across different languages. We assume a generic scenario for *monolingual* users who are interested to find their relevant information under three general settings: (1) find relevant information in a foreign language, which needs machine to translate search results into the user's own language; (2) find relevant information in multiple languages including the source language, which also requires machine translation for back translating search results; (3) find relevant information only in the user's language, but due to the intrinsic cross-lingual nature of many queries, monolingual search can be done with the assistance of cross-lingual information from another language.

We approach the problem by substantially extending two core mechanics of information retrieval for Web search across languages, namely, query formulation and relevance ranking. First, unlike traditional cross-lingual methods such as query translation and expansion, we propose a novel Cross-Lingual Query Suggestion model by leveraging large-scale query logs of search engine to learn to suggest closely related queries in the target language for a given source language query. The rationale behind our approach is the ever-increasing *common search interests* across Web users in *different* languages. Second, we generalize the usefulness of common search interests to enhance relevance ranking of documents by exploiting the correlation among the search results derived from *bilingual queries*, and overcome the weakness of tradi-

tional relevance estimation that only uses information of a single language or that of different languages separately. To this end, we attempt to learn a ranking function that incorporates various similarity measures among the retrieved documents in different languages. By modeling the commonality or similarity of search results, relevant documents in one language may help the relevance estimation of documents in a different language, and hence can improve the overall relevance estimation. This similar intuition is applicable to all the three settings described above.

摘要

利用跨語言的共同搜索興趣幫助萬維網搜索

高巍

香港中文大學博士論文

指導教授：黃錦輝

在這項工作中，我們研究了旨在為滿足用戶的跨語言信息需求的萬維網搜索技術，其中嶄新的研究內容是發現并利用不同語言用戶的共同搜索興趣來提高搜索有效性。我們設定了一個一般性場景，那就是單語言用戶會普遍在如下三種情況下試圖找到他們感興趣的相關內容：（一）找到在一種外語網頁中的相關信息，因此需要機器將搜索結果翻譯成用戶自己的語言；（二）找到在多種外語及原語言中的相關信息，因此同樣需要將外語結果自動翻譯回原語言；（三）只在用戶原語言網頁中找到相關信息，但是由于相當多的查詢具有跨語言的性質，在跨語言信息的幫助下，我們因此可以提高單語言搜索的質量。

我們從查詢表達和相關排序兩個層次來對搜索的核心機制進行重要擴展，以使搜索能夠更好地跨越語言障礙。首先，有別于查詢翻譯和查詢擴展等傳統的跨語言查詢表達方法，我們設計了一個稱為跨語言查詢建議的新模型，它利用大規模搜索引擎的搜索日志為數據來訓練，學習去建議與原語言查詢非常相關的目標語言查詢。此方法背后的基本原理就是利用日益增長的不同語言用戶之間的共同搜索興趣；其次，我們將共同搜索興趣的有效性推廣到文檔的相關排序，它利用了一種稱為雙語查詢的查詢請求所返回的不同語言的搜索結果之間的相關性，這個方法能夠克服傳統的相關性估算方法的缺陷——只考慮使用一種語言的信息或者僅單獨使用不同語言的信息（沒有考慮它們之間的關聯）。為此，我們試圖訓練出一個多語言排序模型，它能夠集成各種用來衡量不同語言返回結果之間相似性的指標。通過對搜索結果的共同屬性或者相似性進行建模，一種語言的相關文檔可以用來幫助估計不同語言文檔的相關性，因此能夠提高總體結果的相關性估算。類似的想法可以應用到前面提到的三種不同的搜索設定之中。

BIBLIOGRAPHIC NOTES

Portions of the dissertation have been published in several conference proceedings and portions will appear in a journal. They are declared as follows:

Chapter 2 is based upon the paper "Cross-Lingual Query Suggestion Using Query Logs of Different Languages", co-authored with Cheng Niu, Jian-Yun Nie, Ming Zhou, Jian Hu, Kam-Fai Wong and Hsiao-Wuen Ho, and has appeared in the *Proceedings of the 30th Annaul International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR'07). A substantially extended version of the paper, titled "Exploiting Query Logs for Cross-Lingual Query Suggestion", has been accepted by and will appear in the *ACM Transactions on Information Systems* (TOIS).

Chapter 3 is based upon the paper "A Joint Ranking Model for Multilingual Web Search", co-authored with Cheng Niu, Ming Zhou and Kam-Fai Wong, and has been published in the *Proceedings of the 31st European Conference on Information Retrieval Research* (ECIR'09). This paper has been awarded "Best Student Paper" at the conference.

Chapter 4 is based upon the paper "Exploiting Bilingual Information to Improve Web Search", co-authored with John Blitzer, Ming Zhou and Kam-Fai Wong, and has been published in the *Proceedings of the 47st Annual Meetings of the Association for Computational Linguistics* (ACL'09).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Web Search across Languages

The growth of the Internet and World Wide Web has been witnessed by the unprecedented popularity of user communication and information dissemination all over the world. In the past decade, the Web has evolved from a network of document repositories serving mainly the research community, where English is the common language, to an universal multilingual platform hosting a wide range of applications accessible by the general public on daily basis, such as electronic commerce, digital library, entertainment, news portal, e-banking, etc. The situation of English language predominating electronic information on the Web has changed dramatically. Web information is now available in an ever-increasing number of languages. In recent years, information is increasingly published in the native language of the provider and can be searched for in the native language of the user. Back in early 2000, Grefenstette and Nioche [32] expected that non-English languages would be growing in a faster rate than English. Consistent to this anticipation, the statistics from website *Internet World Stats* (`www.internetworldstats.com`) shows that compared to the number of English users, the group of non-English users is growing very rapidly in recent years[1]. As of mid-2009, the percentage of non-English Internet users accounts for 71.3% of the world Internet population. This is compatible with another early prediction that by 2005, around 78% of Internet users would be non-English speakers and "only" 49% of Web content would be in English [70].

---

[1] `http://www.internetworldstats.com/stats7.htm`

The explosive growth of the Web has blurred national boundaries to the point where a casual user may find an interest in retrieving documents in a foreign language [52]. While Web search engines become increasingly powerful, they mostly concentrate on searching Web pages in the same language as that of the given query. As a result, relevant pages in other languages are neglected. These pages could be even more relevant to user's information need than those of the original language. For example, English users who like "Peking Roast Duck" may suffer from unknowing the most bargain restaurants before their visit to China. They may be frustrated by the first few retrieved pages returned by an English search engine as this topic is not so popular on English websites. Although the retrieved information is potentially abundant in the Chinese-speaking world, casual English searchers, even if they capable, are reluctant to perform Chinese search due to the difficulty on constructing meaningful Chinese queries and on understanding the search results afterwards. As another example, in the opposite direction, Chinese searchers may want to know some "private aspects of U.S. President Obama", and it makes sense that this interest is more likely to be satisfied by looking for relevant pages from English search results. However, they will have to overcome a similar language barrier.

To meet user's information need across languages, Cross-Language Information Retrieval (CLIR) and Multilingual Information Retrieval (MLIR) are the two traditional search techniques devised to overcome language barrier between users and information. Giving the source language query, CLIR retrieves documents in the collection of target language; and MLIR merges the document lists retrieved from the corresponding collections of source language and different target language(s) into a single reasonable list. Effectively, MLIR can be viewed as combining the results from a monolingual search in the source language and one or more source-to-target CLIR search(es). Once a set of relevant documents in a foreign language is obtained, the user may use automatic machine translation (MT) software to help get some sense out of the content.

Like traditional information retrieval in a single language (i.e., monolingual IR), CLIR and MLIR focus on search within small, controlled and nonlinked document collections, and aim to cater for the needs of a limited number of professional users. In the Web age, the focus of IR research is shifting noticeably towards the search of Web documents to meet the diversity of information needs of the large Internet population at large. Meanwhile, the organizational structure and special properties of Web documents make the Web a unique document collection, which requires systematic studies on search techniques that need to be well tailored to address the new challenges. However, this trend has not been envisaged in Web search across languages due to the technical difficulties one has to encounter to overcome language barriers.

1.2 Technical Challenges

We can resort to CLIR and MLIR techniques to retrieve relevant pages in other languages that are different from query's language. Considered as a whole picture, Figure 1.1 shows an integrated framework including monolingual search, cross-lingual search and multilingual search, where we assume Chinese as the source language and English as the target language (for CLIR and MLIR). In CLIR and MLIR, an input Chinese query q_c is first translated into English q_e. For CLIR, monolingual search is performed in English only, resulting in D_e; and for MLIR, monolingual searches in both source and target languages (i.e., Chinese and English, respectively) are carried out and then both search results in D_c and D_e are merged appropriately. Before presented to the user, English results D_e have to be translated into Chinese $D_{c\leftarrow e}$ for readability. Note that with this figure we attempt to illustrate all the major components of these techniques at a high level and will not focus on an integrated system in this study.

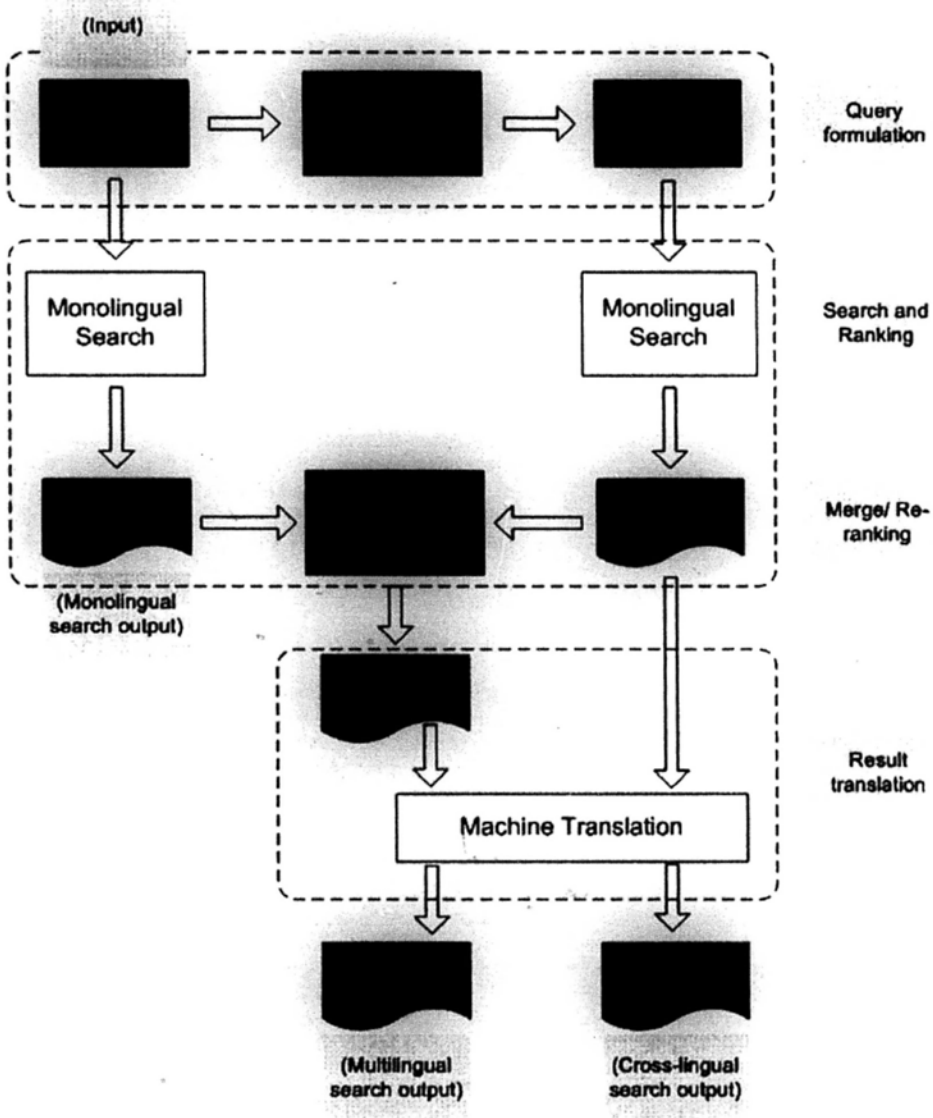

Figure 1.1. The traditional framework of CLIR and MLIR, where we assume Chinese is the source language and English is the target language.

Full-fledged Web search across languages is very challenging due to the difficulties in various aspects of these components: query translation, searching, and ranking or merging of search results in different languages. Machine translation is often necessary to translate the results into user's language. We describe these specific challenging issues from the following three perspectives:

1. The mismatch of keywords in query and documents is the main cause of poor retrieval effectiveness for IR in general. In CLIR and MLIR, the mismatch is harder to deal with due to the existence of language barriers. Traditionally, query translation is used to translate an original query to its counterpart in the target language. The translation can be done by using dictionaries, parallel corpora and existing commercial machine translation (MT) systems. However, these approaches usually rely on static knowledge and data which do not effectively reflect the quickly shifting interests of Web users. One popular query keyword in the source language may be unpopular, or even unknown, in the target language. As a consequence, the translated queries may not be the most reasonable and popular formulations in the target language, even though the terms are reasonable translations of the original terms in the source language query. Therefore, this kind of semantic mismatch may widely exist between the translated terms and those actually used in the target language. Such mismatch makes the translated queries ineffective in retrieving relevant documents in the target language. Direct query translation is also ineffective since accurately translating a query becomes difficult especially for Web queries, which are typically very short. In addition, a large proportion of Web queries contain Out-Of-Vocabulary (OOV) terms, which cannot be translated by traditional techniques at all.

2. In CLIR and MLIR, existing techniques usually combine query translation and monolingual retrieval to derive a relevance score for each document. In CLIR, the relevance ranking is purely monolingual, where the ranking function only takes into account of the query-document relevancy in the target language; and in MLIR, the relevance scores from different CLIR settings are commonly nor-

malized for final combination and ranking. Ranking becomes a difficult task due to the information loss across language boundaries. Due to query translation, the relevance score between the target language query and the documents is somewhat distorted compared to the real document relevancy with the original query. In Web search environment, a wide range of relevance features can be used to compensate this loss of effectiveness. For Web MLIR, for instance, additional relevance features from the source language documents and the correlation features among the retrieved documents can be incorporated in the result merging process. Unfortunately, due to the lack of such studies in the literature, little is known on how to leverage all kinds of features and correlations for improving ranking across languages.

3. Once a set of relevant documents in a foreign language is obtained, monolingual users have to rely on automatic machine translation (MT) software to help get some sense out of the content. Therefore, MT plays an important role since CLIR and MLIR results would be useless to monolingual users who cannot understand them. With the advancement of MT technologies, the readability of translated texts has been improved remarkably in past few years. However, high-quality MT itself is still an open challenge for the research community, and is generally ineffective for general applications. Therefore, in cross-language and multilingual Web searches, it is unrealistic to assume the availability of effective MT technology to facilitate meaningful online browsing.

1.3 Thesis Outline and Contributions

In this dissertation, we aim to address the problems and challenges depicted above and to propose effective cross-lingual and multilingual processing techniques for Web search. In the case where MT cannot come up with readable search results, we will also propose effective means to improve the precision of monolingual search to cater for some important cross-lingual information needs from monolingual users. To deal with all the three challenges, we focus on effective techniques in two critical levels of Web search functionalities, i.e., query formulation and relevance ranking of documents, by using the information derived from common search interests of users across different languages. In query formulation, we intend to resolve the weakness of traditional query translation methods. In relevance ranking, we transcend the limitation of traditional ranking schemes in CLIR and MLIR, where both the relevance features of individual documents and the correlations among documents are leveraged to optimize the ranking. Based on the same intuition of common search interests, we propose different algorithms to improve the effectiveness of relevance ranking for cross-language, multilingual as well as monolingual Web search.

Our study is outlined as below and makes the following contributions:

1. Chapter 2 gives the background and an overall picture of our proposed methodology. We first introduce the intuition of *common search interests* of users across different languages and then presents the statistical properties based on this observed phenomenon. We propose the concept of *bilingual queries* to reflect the common search interests between two groups of users using different languages. It is found that a significant portion of Web information needs fall into this category. We further examine search results of bilingual queries in two languages, and find large qualitative variances between them. These findings provide basic foundations to our techniques proposed in Chapter 3, 4 and 5.

2. Based on the phenomenon of common search interests, Chapter 3 describes a regression-based learning technique on exploiting query logs for *Cross-Lingual Query Suggestion.* We scrutinize the limitations of the traditional query translation and cross-lingual query expansion approaches, and propose a novel method for suggesting closely related queries across languages by exploiting large scale query logs. We make systematic comparisons with traditional query translation

techniques with and without query expansion. When evaluated in TREC and NTCIR CLIR tasks, our method demonstrates obvious advantages.

3. Based on the variance of search quality of bilingual queries, Chapter 4 generalizes the usefulness of underlying common search interests to enhance search result ranking. We present a *joint ranking model* for merging and ranking search results from different languages in multilingual Web search. An effective learning algorithm called *Boltzmann machine* is proposed. The algorithm can take into account of various correlation measures among the retrieved documents in addition to the commonly used query-document relevance features. By using this method, the relevant documents of one language can be leveraged to improve the relevance estimation of documents in different languages, and the joint relevance probability for all the documents can be induced. Compared to various baseline algorithms, our method can significantly improve ranking effectiveness in the TREC CLIR ranking task as well as the multilingual Web search ranking task based on real-world search engine data.

4. In Chapter 5, we propose to improve *monolingual Web search by using bilingual click-through information* derived from common search interests found in the query logs. This technique aims to allow users to meet their important portion of cross-lingual information needs without having to rely on machine-translated search results. For a given bilingual query, with the corresponding monolingual query log and monolingual ranking, we generate a ranking corpus based on pairs of documents, one from each language. We then learn a ranking function by incorporating bilingual features of the document pairs as well as monolingual features of individual documents. Finally, we reconstruct monolingual ranking from a learned bilingual ranking. Using publicly available Chinese and English query logs, we demonstrate that for both languages our ranking technique ex-

ploiting bilingual data leads to significant improvement over the state-of-the-art monolingual ranking algorithm.

CHAPTER 2

BACKGROUND: COMMON SEARCH INTERESTS

The cross-lingual information need has never been so prominent as it is today due to the proliferation of Web content in different languages as well as the rise of common search interests in globally attentive topics among users. On one hand, hot topics can quickly draw worldwide attention due to the fast spread of news and knowledge from different online media; but on the other hand, relevant information for the same information need tends to distribute unevenly among different languages over these media. For these reasons, there is a strong and urgent demand of search technology for relevant information across languages.

2.1 Common Search Interests across Languages

To shed light on the growing common search interests across languages, we start with investigating a particular set of queries, referred to as *bilingual queries*. We designate a query as bilingual if the concept has been searched by two groups of monolingual users, each from one language (e.g., English and Chinese). As a result, not only does it occur in the query log of its own language, but its *translation* also appears in the query log of the second language. For example, "哈利波特" (Harry Potter) and "Peking roast duck" (北京烤鸭) are bilingual queries in Chinese and English source languages, respectively. They are searched by users of both languages, and reflect the common information needs of users using different languages. In this way, a bilingual query yields reasonable queries in both languages. In contrast, *local queries*, such as "长虹电视机" (Changhong TV set) and "PBS Kids" (公共电视网

Table 2.1. Examples of bilingual (first column) and local (third column) queries, together with their translations. Note that the translations of bilingual queries are also bilingual.

Bilingual Chinese query	English translation	Local Chinese query	English translation
福特汽车	Ford Motor	李白写的诗	The poems of Li-Bai
公共关系	public relations	长虹手机	Changhong cell phones
比尔盖茨	Bill Gates	大红鹰	Great Red Eagle
音乐欣赏	music appreciation	碧桂园地产	BiGuiYuan real estate
Bilingual English query	**Chinese translation**	**Local English query**	**Chinese translation**
Jackie Chan	成龙	all black colleges	黑人学院
real estate	房地产	Auto-Locator	汽车搜寻网
China Mobile	中国移动	adoption Tennessee	田纳西收养中心
Honda Civic	本田思域	Anthony Burger	[歌手]安东尼博格

幼儿频道), are not bilingually popular and are most likely to be searched for only in one of the languages. More examples of bilingual and local queries in Chinese and English are given in Table 2.1 together with their corresponding translations.

Of course, most queries are not bilingual. A natural question is why bilingual queries are significant. We did statistics based on two large independent monolingual query logs: AOL log[1] and Sogou log[2]. In total, we extracted over 4.8 million *unique* English queries from AOL log, of which 1.3% of their translations appear in Sogou log; similarly, of over 3.1 million *unique* Chinese queries from Sogou log, 2.3% of their translations appear in AOL log. In terms of unique queries, the proportion of bilingual queries are not large. However, if we count the frequency queries being issued, the proportion of bilingual queries is much higher. Figure 2.1 shows that as the number of times a query is issued increases, so does the chance of it being bilingual. At the highest frequency, in particular, nearly 45% of the English queries and 35% of the Chinese queries are bilingual. This justifies the significance of bilingual queries reflecting the common information needs of English and Chinese users.

[1]`http://search.aol.com`, English query log from AOL search engine

[2]`http://www.sogou.com`, Chinese query log from Sogou search engine

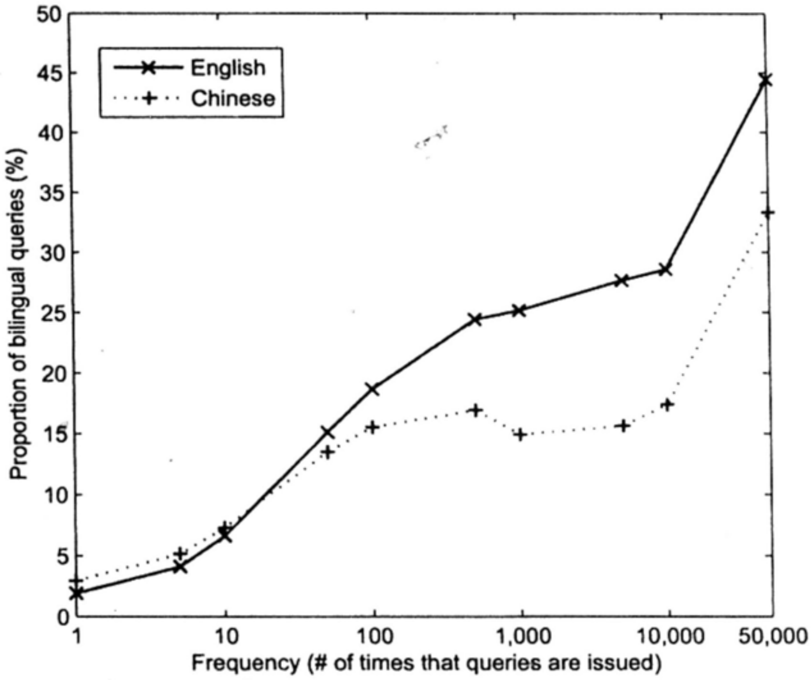

Figure 2.1. With the increase of frequency, proportion of bilingual queries increases in the query logs of different languages.

We should highlight two additional factors related to our statistics above that can further strengthen our view on the significance of bilingual queries:

1. We originally defined bilingual query in a rather strict manner, i.e., a bilingual query and its counterpart in the other language should be queries that are *accurately* and *mutually* translatable. This is primarily for the purpose of simplifying our statistical calculation. In reality, users of individual languages issue their queries in personal and independent fashions. Thus the same information need may be represented using diverse formulations of queries by different users. For example, Chinese users commonly search for "Jennifer Aniston" by issuing "安妮斯顿", her surname in Chinese, as a query; most English users, however, like to search by her full name (in English) instead. By counting this kind of cross-lingual relatedness or similarity, the proportion of bilingual queries in a

broad sense would be much higher, implying large commonalities among search interests across the languages.

2. We conducted statistics on bilingual queries between English and Chinese, where the two languages are considered less correspondent. Chinese is considered more weakly correlated with English than other western languages, e.g., French. It is reasonable to conjecture that there is stronger correspondence between the query logs in closely correlated languages. Due to the close social and cultural background among people of those languages, their common search interests tend to be greater than those of people from communities which are linguistically, geographically, and culturally apart. Therefore, bilingual queries in former kind of languages are more readily available.

Using monolingual query logs of different languages, we can find that the users of monolingual search engines in the same period of time have common search interests, and they submit queries on similar topics in their own languages. As a result, a query written in a one language likely has a correspondent in a query log in the other language. In particular, note that if the user intends to perform cross-language search, he/she tends to retrieve something popular or well-known in the target-language domain. Thus, it turns out that such an original query is even more likely to have its correspondent included in the target-language query log. Potentially, this kind of commonality can play a crucial role for meeting user's cross-lingual information needs. Furthermore, if the click-through data associated with these queries are taken into consideration, the sources of information originated from common search interests will become even more extensive. Throughout this dissertation, we will investigate and generalize the usefulness of common search interests across languages for effective Web information retrieval.

Table 2.2. The uneven distribution of relevant information across different languages in some common topics (estimated based on search results of Google, as of November 2009).

Chinese topic	# of pages	Corresponding English topic	# of pages
气球男孩事件	1,760,000	Colorado balloon incident	2,610,000
美国军营枪击案	183,000	Texas military base shooting	5,910,000
比尔盖茨	2,650,000	Bill Gates	22,700,000
迈克尔杰克逊去世	516,000	Michael Jackson Death	79,400,000
美国偶像	7,060,000	American Idol	34,000,000
四川地震	23,000,000	Sichuan earthquake	1,310,000
百度	182,000,000	Baidu	31,000,000
孔子	9,100,000	Confucius	3,410,000
北京奥运会	15,300,000	Beijing Olympic games	2,490,000
北京烤鸭	14,400,000	Peking roast duck	74,800

2.2 Variance of Search Quality in Different Languages

Common search interest between languages is an important indicator that users from both sides have strong information need on a topic. However, it is frequent that Web search quality may vary widely across different languages even for the same information need. For example, search result for the query "托马斯霍布斯" (Thomas Hobbes, an English philosopher) is relatively poorer in Chinese than it is in English because there is less relevant information on Chinese Web due to the English origin of this query; on the other hand, search results for "The Duke of Zhou" (周公, an ancient Chinese politician) is worse in English than in Chinese due to its Chinese origin. As an illustration, Figure 2.2 and Figure 2.3 show the details of the top search results of "Thomas Hobbes" in Chinese and English, respectively, returned from a commercial search engine[3]. As we can see, four of top-5 English results are relevant whereas all the top-5 Chinese results are not.

In addition, we observe that the amount of relevant information (approximated by the number of retrieved pages) for bilingual queries is often unevenly distributed in different sides: the number of relevant pages in the original language of the queried

[3] `http://www.bing.com`

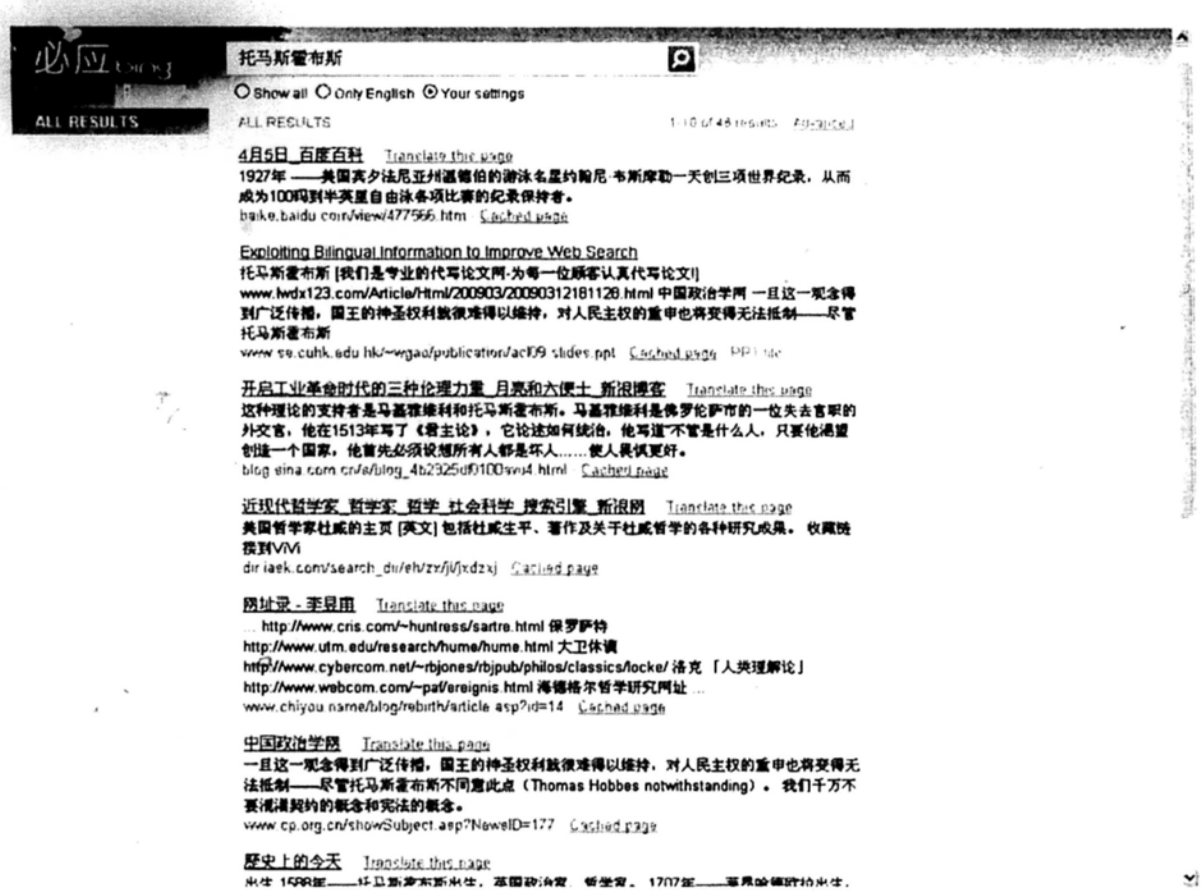

Figure 2.2. Chinese search results for the bilingual query "托马斯霍布斯", where none of the top-5 results is relevant.

concept are typically much larger than the number of pages in other languages. Table 2.2 illustrates our observation, which shows the numbers of retrieved result items from Google for some topics widely recognized in the English and Chinese worlds. As we can see, the number of retrieved pages of the topics originated from the English domain is obviously larger than that of the corresponding topics translated to Chinese, and vice versa. Although not directly reflecting relevancy, it does reflect the discrepancy of search quality due to the amount of information available in each side. Therefore, it makes sense to investigate the methods of using this variance of quality to improve search result in different languages.

Information on the Web and information need of users are of interdependence. That is, with only few queries about something in a language by large means little

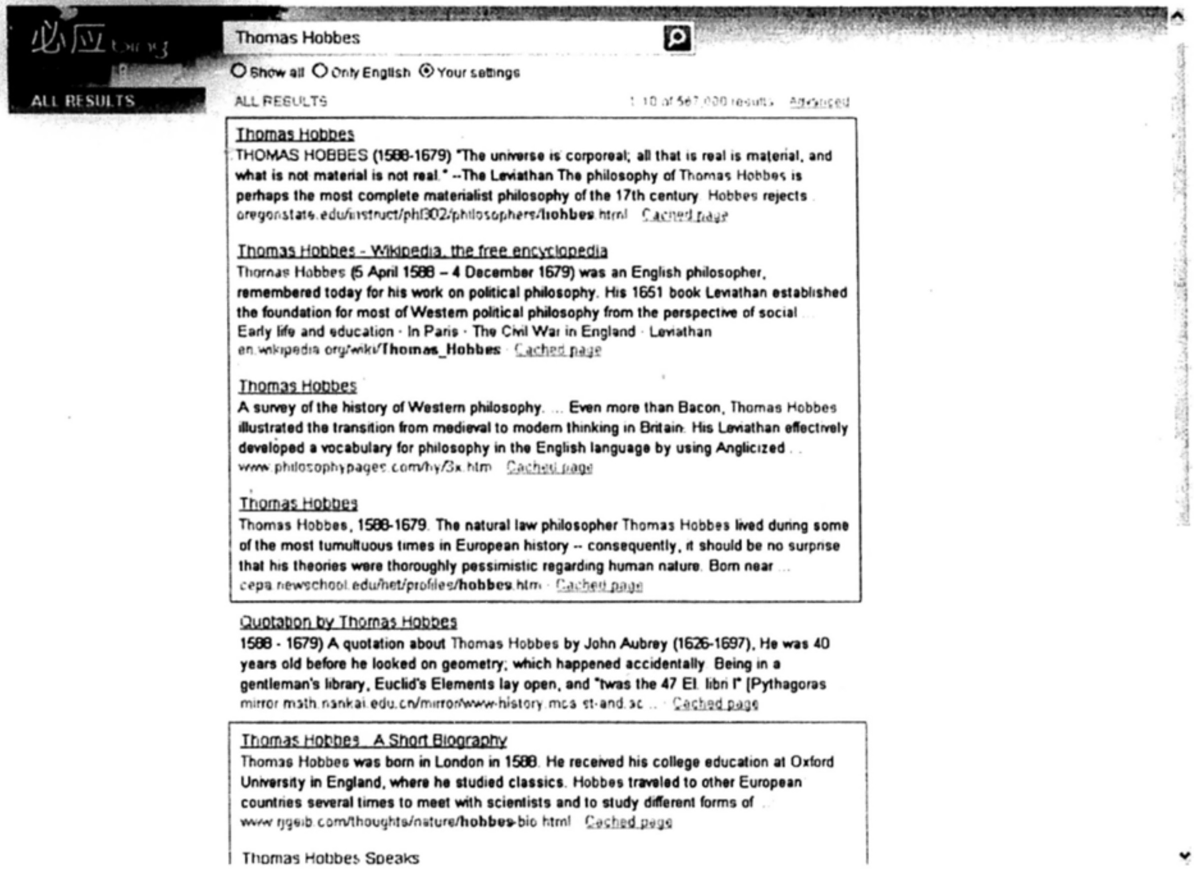

Figure 2.3. English search results for the bilingual query "Thomas Hobbes", where most of the top results are relevant.

relevant information is available in that language about the topic, and vice versa. For local queries, due to the lack of exposure to outside world, it can be naturally conjectured that relevant information would be distributed even more unevenly between local and foreign languages than it is for bilingual queries. Although local queries have no correspondent queries in query log of a foreign language, they still can be translated to become a new query in that language. Therefore, these new queries, although rare topics in the "new" language, once needed by some monolingual foreign users, would be suitably handled by cross-language search for finding their relevant documents on the Web of their original language.

2.3 Proposed Methods

Based on the observation and analysis on common search interests of users across different languages, we propose the core Web search techniques described below. Note that our methods are generic and language-independent, i.e., there is nothing specific to our method that is dependent on the source or target languages and the number of languages involved, as long as the required data resources are available.

Cross-Lingual Query Suggestion — In order to address the keyword mismatch in CLIR, we propose the Cross-Lingual Query Suggestion (CLQS) technique using query logs of different languages. At the level of query formulation, CLQS automatically suggests closely related queries in the target language instead of pursuing accurate query translation like most existing approaches in the literature. We argue that accurate query translation is neither necessary, nor sufficient, for CLIR. Based upon our observation on common search interests among search users of different languages, the innovative thrust of CLQS is that by mining large-scale query logs of search engine, we can discover common query formulations in the target language, which are highly related to the semantics of the original information need. By making use of up-to-date query logs from the target site, it is expected that for most user queries, we can find common formulations on these topics in the query log of the target language. Therefore, CLQS plays a role of adapting the original query formulation to the common formulations of similar topics in the target language. Query log also bears good coverage of queries. For this reason, unknown query words, which do not exist in the bilingual resources can likely be recovered from the query logs.

Joint Ranking for CLIR and MLIR — The information loss caused by query translation makes CLIR and MLIR ranking of documents a difficult task. We generalize the usefulness of common search interests to enhance relevance ranking by exploiting the correlation among search results derived from *bilingual* *queries*. To complement the distorted query-document relevancy, we propose a *joint ranking model* to incorporate inter-document similarities. If two documents are bilingually correlated or similar, and one of them is relevant to the query, it is very likely that the other is also relevant. By modeling the similarity among search results, relevant documents in one language can be leveraged to help the relevance estimation of documents in different languages, and hence can improve the overall relevance estimation. This special form of pseudo feedback is done by leveraging the variance of search quality in different languages (see Section 2.2). Unlike existing approaches that are focused on combining relevance scores of different result lists, we learn a multilingual ranking function directly by incorporating various relevance features of retrieved documents. This is advantageous in that the optimal combination of features and similarities can be solved using the popular learning-to-rank formalism, which provides solid mathematical ground to the final relevancy scores of the documents.

Monolingual Ranking Using Cross-lingual Information — The previous two objectives proposed above are focused on effective cross-language and multilingual search based on the assumption that machine translation can help users smoothly understand search results in foreign languages. However, this assumption is too strong to be realistic giving MT's unsatisfying effectiveness at present time. For this reason, we simply *turn to study the possibility of enhance monolingual search to meet user's important cross-lingual information needs*. The intuition behind is based upon our observation that search quality varies widely across different languages even for the same information needs, and thus we may exploit search ranking of one language and cross-lingual related documents to help the ranking in another language. It is well-known that the precision at top few positions are of paramount importance in Web search. Using our method, monolingual search can be tailored in a way to enhance the precision of top search results for those queries whose retrieved relevant documents are ordered behind the top few positions in the ranked list. Although our method may not be universally effective for all queries, we will concentrate on improving the ranking quality of bilingual queries that represent a large proportion of cross-lingual information needs. Therefore, we will be able to eliminate the requirements on MT and at the same time to better meet the important portion of cross-lingual search interests.

2.4 Chapter Summary

This chapter depicts the background, on which our techniques are proposed for improving Web search. Common search interests of users across different languages are important cross-lingual information needs as shown by our preliminary studies on bilingual queries and their search results with variance of qualities in different languages. These findings motivate us to provide three aspects of technologies based on the observed phenomenon and statistics to enhance Web search.

CHAPTER 3

CROSS-LINGUAL QUERY SUGGESTION (CLQS)

Query suggestion aims to suggest relevant queries for a given query, and helps users better specify their information needs. Previous work on query suggestion has been limited to the same language. We extend it to cross-lingual query suggestion. For a query in one language, we suggest similar or relevant queries in other languages. This is very important to Cross-Language Information Retrieval (CLIR) and related applications. Instead of using existing query translation technologies for CLQS, we propose an effective means to map an input query of one language to queries of another language in the query log. Important monolingual and cross-lingual information such as word translation relations and word co-occurrence statistics, etc., are used to estimate the cross-lingual query similarity with a discriminative model. Benchmarks show that the resulting CLQS system significantly outperforms a baseline system using dictionary-based query translation. Besides, we evaluate CLQS with French-English and Chinese-English CLIR tasks on TREC-6 and NTCIR-4 collections, respectively. The CLIR experiments using typical IR models demonstrate that the retrieval effectiveness based on our CLQS approach is significantly higher than several traditional query translation methods. Moreover, we find that combining CLQS with pseudo-relevance feedback can further improve retrieval effectiveness for different language pairs.

This chapter is based on our work published in [28, 29].

3.1 Introduction

Query suggestion is a designed to help users of a search engine better specify their information needs. This is accomplished by narrowing down or expanding the scope of a search with synonymous queries and relevant queries, or by suggesting related queries that have been frequently used by other users. Popular search engines, such as *Google*[1], *Yahoo!*[2], *Live Search*[3], *Ask.com*[4], do provide query suggestion functionality as a valuable addition to their core search technology. Moreover, the same approach has been applied to recommend bidding terms to online advertisers in the pay-for-performance search market [31].

Query suggestion is related to query expansion which extends the original query with new search terms to narrow down the scope of the search. But different from query expansion, query suggestion aims to suggest full queries that have been formulated by users so that the query integrity and coherence are preserved in the suggested queries. Therefore, it is expected to play an alternative or complementary role to query expansion in information retrieval applications.

Typical methods for query suggestion exploit query logs and document collections, by assuming that in the same period of time, many users share the same or similar interests, which can be expressed in different manners [31, 39, 97]. By suggesting the related and frequently used formulations, it is expected that the new queries can cover more relevant documents.

To our knowledge, all existing studies only deal with monolingual query suggestion. There is no research on cross-lingual query suggestion (CLQS) by exploiting query logs. CLQS aims to suggest related queries in a different language. It has

---

[1] `http://www.google.com`

[2] `http://search.yahoo.com`

[3] `http://www.live.com`

[4] `http://www.ask.com`

important applications on the World Wide Web such as cross-language search or suggesting relevant bidding terms in e-advertising.

CLQS can be approached as a query translation problem, i.e., to formulate the queries that are translations of the original query. Dictionaries, large-size parallel corpora and existing commercial machine translation (MT) systems can be used for translation. However, these kind of approaches usually rely on static knowledge and data that cannot effectively reflect the quickly shifting interests of Web users. As a consequence, even though the terms can be reasonable translations of the original terms in the source language query, the suggested queries may not be the most reasonable and popular formulations in the target language. For example, the French query "aliment biologique" is translated into "biologic food" by Google's machine translation tool[5]. At the term level, the translation seems reasonable. However, the correct formulation should be "organic food". Similarly, the Chinese query "动物复制" is translated literally as "animal reproduction", but in fact it is widely expressed as "animal cloning" in English. There are many such mismatch cases between the translated terms and those actually used in the target language. Such mismatches render the translated queries ineffective in finding relevant documents in the target language.

A natural way in solving this mismatch problem is to exploit query logs in the target language to select the most popular query formulations corresponding to the original query in the source language. Ideally, the selection would be most effective if one has a query log with aligned queries between the source and target languages. However, such a resource does not exist. In practice, we only have separate query logs in source and target languages for the same period of time. Such resources are still very useful to us. We found that the two separate query logs cover many common

[5]`http://www.google.com/translate_t`

search interests (see Section 2.1). Therefore, it can be expected that for many queries in the source language we can find their corresponding or similar queries in the target-language query log, especially for popular queries.

The query logs can be used in the following way for CLQS: when a source-language query is submitted, we try to determine the most similar query in the target-language query log. In addition to considering the translation relation between the source-language query and the target-language suggestions, we also leverage the following two effects from the target-language query log:

1. The suggested queries from the target-language query log are complete queries, which correspond to the normal ways users formulate queries in the target language. In this way, compared to the translation approach, more natural formulation of queries can be obtained.

2. The suggested queries from the target-language query log can not only be the translation of the original query, but also strongly related ones. Therefore, we can more naturally achieve the desired effect of query expansion to reflect users' needs.

A key issue to arrive at reasonable cross-lingual query suggestion is the estimation of cross-lingual query similarity. We propose a new method for calculating this similarity by exploiting, in addition to the translation information, a wide spectrum of bilingual and monolingual information, such as term co-occurrences, query logs with click-through data, etc. A discriminative model is used to learn the calculation of cross-lingual query similarity based on a set of manually translated queries. The model is trained by optimizing the cross-lingual similarity to best fit the monolingual similarity between one query and the other query's translation.

The resulting CLQS system is evaluated as an independent module as well as a new means of query "translation" for French-English and Chinese-English CLIR tasks using prevalent retrieval models based on TREC-6 and NTCIR-4 data collections, respectively. It is then compared with several traditional query translation methods including a dictionary-based translation approach using co-occurrence-based translation disambiguation, a phrase-based statistical machine translation (SMT) system, and an automated translation extraction technique by mining unknown query translations from Web corpora. The results show that this new "translation" method is more effective than the other approaches. Furthermore, we show that when combined with pseudo-relevance feedback (PRF), CLIR effectiveness is further improved.

This chapter is organized as follows: Section 3.2 presents the related work; Section 3.3 describes in detail the discriminative model for estimating cross-lingual query similarity; Section 3.4 presents a new CLIR approach using cross-lingual query suggestion as a bridge across language boundaries. Section 3.5 discusses the experiments and results; finally, we summarize this chapter in Section 3.6.

3.2 Related Work

Most approaches for CLIR are achieved by query translation followed by monolingual IR. Typically, queries are translated using a bilingual dictionary [71], a machine translation system [23], a parallel [66] or comparable corpus [60].

Despite the various types of resources used, out-of-vocabulary (OOV) words and translation disambiguation are the two major bottlenecks for CLIR [66]. In [16, 105], OOV term translations were mined from the Web using a search engine. In [61], bilingual knowledge was acquired based on anchor text analysis. In addition, word co-occurrence statistics in the target language were applied for translation disambiguation [4, 26, 25, 65].

When query translation is employed for CLIR, Kwok et al. [51] utilized translation results from different MT tools and translation resources. The system achieved better CLIR effectiveness than the single translation approach. Although we also resort to various translation resources, our CLQS approach is different from Kwok's in that we employ the resources for finding relevant candidate queries in the query log rather than for acquiring accurate translations.

It is arguable that accurate query translation may neither be necessary, nor sufficient, for CLIR. In many cases, it is helpful to introduce words that are not direct translations of any query word, but are closely related to the meaning of the query. From a translation point of view, such a translation is certainly imperfect. However, several experiments have shown that such a translation could perform better than a high-quality MT result [49], and even better than a professional manual translation for CLIR purpose[26]. This observation has led to the development of cross-lingual query expansion (CLQE) techniques [3, 52, 64]. Ballesteros and Croft [3] reported the enhancement on CLIR by post-translation expansion. Lavrenko [52] developed a cross-lingual relevancy model by using the cross-lingual co-occurrence statistics in parallel texts. McNamee and Mayfield [64] compared the performance of multiple CLQE techniques, including pre- and post-translation expansions. However, a unified framework to combine the wide range of resources and Web mining techniques for CLQE is yet unavailable.

López-Ostenero [60] proposed method for cross-language search by accurate translation of the noun phrases in a query, followed by a blind expansion with frequent phrases. Their bilingual phrase alignment dictionary was built on a comparable corpus, and query refinement is fulfilled by using the phrase-based summary of document content. This technique could be considered as a noun-phrase-based CLQE.

CLQS is different from CLQE in that it aims to suggest full queries that have been formulated by users in another language. Our CLQS approach exploits up-to-date query logs. It is expected that for most user queries, we can find common formulations on these topics in the query log of the target language. Therefore, CLQS also plays a role of adapting the original query formulation to the common formulations of similar topics in the target language.

Query logs have been successfully used for monolingual IR, especially in monolingual query suggestions [31] and in relating semantically relevant terms for query expansion [18, 41]. In [2], the target language query log has been exploited for query translation in CLIR. White et al. [98] compared the similarity of refined queries using query logs and PRF in Web search. Based on a BM25 retrieval model [79], our recent work [28] showed that in the French-English CLIR task, a CLQS-based approach outperformed dictionary-based method and an online MT tool from Google for query translation, and the combination of CLQS and PRF could be complementary and improved CLIR effectiveness.

Nevertheless, several important issues remain unclear and unexplored in our previous study: (1) When queries are translated using online MT software such as Google, it is difficult to compare it with CLQS because the translation quality frequently changes due to product updates made by the service provider. In addition, the techniques and data resources used for constructing the MT system are unknown to us; (2) It is unclear how CLQS-based CLIR performs compared to query translation under different IR frameworks, especially when PRF is introduced. Since PRF techniques vary with the underlying retrieval models, it is uncertain whether PRF could consistently complement to CLQS; (3) It is unknown if high-quality queries could be suggested using query logs across linguistically dissimilar languages, such as Chinese-English. It is interesting to investigate the effectiveness of CLQS for such a language pair where the correspondence between users' search interests might not be so strong.

3.3 Estimating Cross-lingual Query Similarity

A search engine has a query log containing user queries with time stamps. In addition to queries, click-through information is also recorded. Therefore, we know which documents have been selected by users for each query. A search engine is used simultaneously by users in different languages, or more precisely, each version of the search engine is used by users of a language group (and locale). We then have a query log for each language (or locale) at the same time period. The simultaneous query logs are the key resources that we exploit in this study. Given a query in the source language, our CLQS task is to determine one or more similar queries in the target language from the query log.

The key problem with cross-lingual query suggestion is how to learn a similarity measure between two queries in different languages. Although various statistical similarity measures have been proposed for monolingual terms [18, 97], most of them were based on term co-occurrence statistics, and could hardly be applied directly to cross-lingual settings since terms of different languages are not so likely to co-occur as monolingual terms.

In order to define a similarity measure across languages, one has to use at least one translation tool or resource. As such, the measure will be based on both translation relation and monolingual similarity. In this work, we aim to provide up-to-date query similarity measure, and static translation resources alone is not sufficient. Therefore, we also integrate a method to mine possible translations from the Web. This method is particularly useful for dealing with OOV terms.

Given a set of resources of different natures, the next question is how to integrate them in a principled manner. In this study, we propose a discriminative model to learn the appropriate similarity measure. The principle is as follows: we assume that we have a reasonable monolingual query similarity measure. For any training query example for which a translation exists, its similarity measure (with any other query) is transposed to its translation. Therefore, we have the desired cross-language similarity value for this example. We then use a discriminative model to learn the cross-language similarity function which best fit these examples.

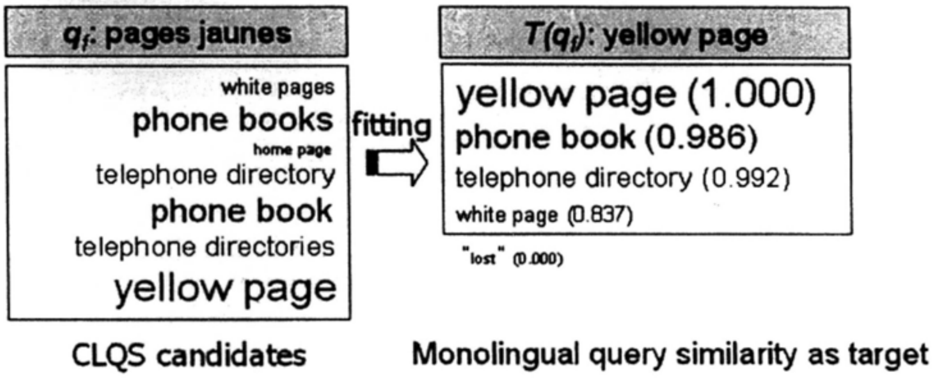

Figure 3.1. An illustration of the principle to transpose cross-lingual query similarity to monolingual query similarity for CLQS candidates to fit as target values. Note that the matched queries are displayed with the characters in the same size.

In the following sections, the detail of the discriminative model for cross-lingual query similarity estimation is described. We then introduce all the features (monolingual and cross-lingual information) which will be used in the model.

3.3.1 Discriminative Model

We first assume a reasonable monolingual query similarity measure as the target in the discriminative training. For a pair of queries in different languages, their cross-lingual similarity should fit the monolingual similarity between one query and the other query's translation. For example, the similarity between French query "pages jaunes" (i.e., "yellow pages" in English) and English query "telephone directory" should be equal to the monolingual similarity between the translation of the French query "yellow page" and "telephone directory". Figure 3.1 shows an illustration of our principle based on this example.

Compared to a query translation approach, the above approach has several advantages:

1. Monolingual query similarity can be estimated more accurately than cross-lingual query similarity, and there are many ways and resources available for

it. Using our approach, we can take advantage of the monolingual similarity to deduce a way to estimate cross-lingual query similarity.

2. Cross-lingual query suggestion is not limited to query translation. Similar queries in the target language can also be suggested, even though they are not direct translations. For example, "telephone directory" can be suggested for the French query "pages jaunes". This will naturally lead to the effect of query expansion.

3. The suggested queries in the target language are those that appeared frequently in the query logs in the target language. Thus, we can also take into account the way that queries are formulated by users in the target language. For example, if the query "organic food" is submitted much more often than the query "biologic food" in English, then the former would be suggested for the French query "nourriture biologique".

The target monolingual query similarity can be determined in various ways, e.g., using term co-occurrence based mutual information [40] and chi-square [16]. Any of them can be used as the target for the cross-lingual similarity function to fit. In this way, cross-lingual query similarity estimation is formulated as a regression task described below.

Given a source language query q_f, a target language query q_e, and a monolingual query similarity sim_{ML}, the corresponding cross-lingual query similarity sim_{CL} is defined as the following:

$$sim_{CL}(q_f, q_e) = sim_{ML}(T_{q_f}, q_e) \tag{3.1}$$

where T_{q_f} is the translation of q_f in the target language.

Based on Equation 3.1, it would be relatively easy to create a training corpus. All it requires is a list of query translations compiled by human experts and a monolingual query similarity function. An existing monolingual query suggestion system can then be used to automatically produce similar queries to each translation, and create the training corpus for cross-lingual similarity estimation; Another advantage is that it is fairly easy to make use of arbitrary information sources within a discriminative modeling framework to achieve optimal performance.

In this work, support vector machine (SVM) regression algorithm [89] is used to learn the cross-lingual term similarity function. Given $\mathbf{f}$, a vector of feature functions with respect to q_f and q_e, $sim_{CL}(q_f, q_e)$ is represented as an inner product between a weight vector and the feature vector in a kernel space as follows:

$$sim_{CL}(q_f, q_e) = \mathbf{w} \cdot \phi(\mathbf{f}(q_f, q_e)) \tag{3.2}$$

where $\phi(.)$ is the mapping from the input feature space onto the kernel space, and $\mathbf{w}$ is the weight vector in the kernel space which will be learned by the SVM regression training. Once the weight vector is learned, Equation 3.2 can be used to estimate the similarity between queries of different languages.

It is noteworthy that instead of regression, one can simplify the learning process as a binary or ordinal classification task, in which case CLQS can be categorized according to discontinuous class labels, e.g., *relevant* and *irrelevant*, or a series of levels of relevancies, e.g., *strongly relevant*, *weakly relevant*, and *irrelevant*. In either case, one can resort to discriminative classification approaches, such as an SVM or maximum entropy model, in a straightforward way. However, the regression formalism enables us to fully rank the suggested queries based on the similarity score given by Equation 3.1.

Equations 3.1 and 3.2 construct a regression model for cross-lingual query similarity estimation. In the following sections, the monolingual query similarity measure (see Section 3.3.2) and the feature functions used for SVM regression (see Section 5.3) are presented.

3.3.2 Monolingual Query Similarity Measure

Any monolingual term similarity measure can be used as the regression target. We adopt the monolingual query similarity measure presented in [97], which used search users' click-through information in query logs and performed effectively in monolingual query suggestion. The reason to choose this monolingual similarity measure is that it is defined in a context similar to ours, i.e., according to a user log that reflects users' intention and behavior. Therefore, we can expect that the cross-lingual query similarity learned from it can also reflect users' intention and expectation.

Following [97], our monolingual query similarity is defined by combining both query content-based similarity and click-through commonality in the query log. First, the content similarity between two queries p and q is defined as follows:

$$similarity_{content}(p,q) = \frac{KN(p,q)}{\max(kn(p), kn(q))} \tag{3.3}$$

where $kn(x)$ is the number of keywords in query x, $KN(p,q)$ is the number of common keywords in the two queries. Secondly, the click-through-based similarity is defined as follows:

$$similarity_{click\text{-}through}(p,q) = \frac{RD(p,q)}{\max(rd(p), rd(q))} \tag{3.4}$$

where $rd(x)$ is the number of clicked URLs for query x, and $RD(p,q)$ is the number of common URLs clicked for the two queries. These two similarity measures represent different points of views. The content-based measure captures queries with the same or similar terms without considering semantic relatedness, such as "Barack Obama", "Obama Barack", "Senator Barack Obama", etc., and the click-through-based measure captures queries semantically related to the same or similar topics, such as "Illinois Senator", "Obama 2004 democratic national convention", "Michelle Obama", etc.

However, user's information need may only be partially captured by either of the measures. In order to take advantage of both strategies, the similarity between two queries can be formulated as a linear combination of the two similarities, which is presented as follows:

$$sim_{ML}(p,q) = \delta * similarity_{content}(p,q) + (1-\delta) * similarity_{click\text{-}through}(p,q) \quad (3.5)$$

where δ is the relative importance of the content-based similarity. In this work, we set $\delta = 0.4$ empirically. If a query p has a similarity measure higher than a certain threshold with another query q, q will be regarded as a relevant monolingual query suggestion (MLQS). The threshold is set as 0.9 empirically. Note that [97] described details about parameter tuning and the impact of the threshold on MLQS.

3.3.3 Features for Learning Cross-Lingual Query Similarity

This section presents the extraction of candidate relevant queries from the log with the assistance of various monolingual and bilingual resources. Also, feature functions over source query and the cross-lingual relevant candidates are defined. Some of the resources being used here, such as bilingual lexicon and parallel corpora, were widely used for query translation in previous work [4, 26, 64, 66, 71]. But note that we employ them for a different purpose, i.e., for finding relevant candidates in the log rather than for acquiring accurate translations.

3.3.3.1 Bilingual Dictionary

In this subsection, we present how a bilingual dictionary is used to retrieve candidate queries from query log. Since multiple translations may be associated with each source word, co-occurrence-based translation disambiguation [4, 26, 25] is performed and described below.

Given an input query $q_f = w_{f1}w_{f2}\dots w_{fn}$ in the source language, for each query term w_{fi}, a set of unique translations provided by the bilingual dictionary is denoted as $T_i : D(w_{fi}) = \{t_{i1}, t_{i2}, \dots, t_{im}\}$. We then try to determine a measure of cohesion between the translations of different query words w_{fi} and w_{fk} $(i \neq k)$. A cohesive query is the one that has a high likelihood to be formed in the target language. Here, we define the cohesion between the translation terms of two query terms, i.e., $t_{ij} \in T_i$ and $t_{kl} \in T_k$ ($T_k : D(w_{fk}) = \{t_{k1}, t_{k2}, \ldots, t_{km}\}$ is a set of translations of term w_{fk}), according to the following mutual information (MI) formula:

$$MI(t_{ij}, t_{kl}) = P(t_{ij}, t_{kl}) log \frac{P(t_{ij}, t_{kl})}{P(t_{ij})P(t_{kl})} \tag{3.6}$$

where $P(t_{ij}, t_{kl}) = \frac{C(t_{ij}, t_{kl})}{N}$ and $P(t) = \frac{C(t)}{N}$. Here $C(x, y)$ is the number of queries in the log containing both x and y, $C(x)$ is the number of queries containing term x, and N is the total number of queries in the log. The MI value indicates how likely two translation terms co-occur in the queries of the target-language log.

Based on the term-term cohesion defined in Equation 3.6, the optimal set of query translations can be approximated with a greedy algorithm in [26] to select the word in each T_i which has the highest degree of cohesion with the translation words in other set T_k. The set of best words from each translation set forms our query translation T'_{q_f} measured by the summation of the term-term cohesion:

$$S_{dict}(T'_{q_f}) = \sum_i \max_{ij} \sum_{k, k \neq i} \max_{kl} MI(t_{ij}, t_{kl}) \tag{3.7}$$

The algorithm then iteratively finds the next set of best translation words by excluding one or more of the selected words. All the generated query translations are added into the set $\{T'_{q_f}\}$ and ranked by $S_{dict}(T'_{q_f})$ score. For each query translation $T \in \{T'_{q_f}\}$, we retrieve all the queries containing the same keywords as T from the target-language log. The retrieved queries are candidate target queries, and are assigned $S_{dict}(T)$ as the value of the feature *Dictionary-based Translation Score*. By trial and error on different number of candidates, we empirically select 4 best candidate target queries ranked by $S_{dict}(T)$ score for the suggestion, which yield nearly optimal training performance. The number of candidates is also determined in a similar way for candidate extraction using parallel corpora and Web mining.

3.3.3.2 Parallel Corpora

Parallel corpora are precious resources for bilingual knowledge acquisition. Different from the bilingual dictionary, the bilingual knowledge learned from parallel corpora assigns probability for each translation candidate which is useful in acquiring dominant query translations.

A parallel corpus is first aligned at sentence level. Word alignments can then be derived by training an IBM translation model-1 [7] using *GIZA++* [68]. The learned bilingual knowledge is used to extract candidate queries from the query log.

Given a pair of queries, q_f in the source language and q_e in the target language, the *Bi-Directional Translation Score* is defined as follows:

$$S_{model\text{-}1}(q_f, q_e) = \sqrt{P_{model\text{-}1}(q_f|q_e) \times P_{model\text{-}1}(q_e|q_f)} \tag{3.8}$$

where $P_{model\text{-}1}(y|x)$ is the word sequence translation probability given by IBM model-1 which has the following form:

$$P_{model\text{-}1}(y|x) = \frac{1}{(|x|+1)^{|y|}} \prod_{j=1}^{|y|} \sum_{i=0}^{|x|} P(y_j|x_i) \tag{3.9}$$

where $P(y_j|x_i)$ is the word-to-word translation probability derived from the word-aligned corpora.

The reason to use bidirectional translation probability is to deal with the fact that common words can be considered as possible translations of many words. By using bidirectional translation, we test whether the translation words can be translated back to the source words. This is helpful to enhance the translation probability of the most specific translation candidates.

Now given an input query q_f, the top-10 queries $\{q_e\}$ with the highest bidirectional translation scores with q_f are retrieved from the query log, and $S_{model\text{-}1}(q_f, q_e)$ in Equation 3.8 is assigned as the value for the feature *Bi-Directional Translation Score*.

3.3.3.3 Web Mining for Related Queries

The translation of unknown words or Out-Of-Vocabulary (OOV) words is a major knowledge bottleneck for query translation and CLIR. To overcome this predicament, Web mining has been exploited in [16, 105] to acquire English-Chinese term translations. The proposed methods are based on the observation that Chinese terms may co-occur with their English translations, for example, "...皇家马德里 (Real Madrid)..." in the same Chinese Web page. This approach works well for foreign proper names that occur frequently in Web pages. Our goal is broader. We are not limited to mining translations of unknown words; instead we are also interested in mining strongly related terms. For example, we expect the queries relevant to "贝克汉姆" (David Beckham) to be mined as well for this example as this proper name is very likely to occur within the context of the Web pages and/or query logs related to Real Madrid. In this section, we describe a variant of this approach to acquire both translations and semantically related queries in the target language.

It is assumed that if a query in the target language co-occurs with the source query in many Web pages, they are probably semantically related. Therefore, a simple method is to send the source query to a search engine (e.g., Google) for Web pages in the target language in order to find related queries in the target language. For instance, by sending a French query "pages jaunes" to search for English pages, the English snippets containing the key words "yellow pages" or "telephone directory" will be returned. However, this simple approach may induce significant amount of noise due to the non-relevant returns from the search engine. In order to improve the relevancy of the bilingual snippets, we extend the simple approach by the following query modification: the original query is used to search with the dictionary-based query keyword translations, which are unified by the $\wedge$ (AND) and $\vee$ (OR) operators into a single Boolean query. For example, for a given query $q = abc$ where the set of translation entries in the dictionary for word a is $\{a_1, a_2, a_3\}$, b is $\{b_1, b_2\}$ and c is $\{c_1\}$, we issue $q \wedge (a_1 \vee a_2 \vee a_3) \wedge (b_1 \vee b_2) \wedge c_1$ as one Web query.

From the top 700 returned snippets of each constructed Boolean query, query translations are first identified using the *SCPCD* (Symmetric Conditional Probability with Context Dependency) measure from all word n-grams in the target language. *SCPCD* combines the symmetric conditional probability (*SCP*) with the context dependency (*CD*) for n-grams, and is used as an association measure for determining an n-gram as well-formed phrase (see [16] for details). The most frequent 10 candidate queries are then retrieved from the query log and are associated with the features of *Frequency in the Snippets*.

Furthermore, we use Co-Occurrence Double-Check (*CODC*) measure to weight the relatedness between the source and target queries. *CODC* measure is proposed in [15] as an association measure based on snippet analysis, referred to as the Web Search with Double Checking (*WSDC*) model. In *WSDC* model, two objects a and b are considered to have an association if b can be found by using a as query (forward process), and a can be found by using b as query (backward process) in the Web search. The forward process counts the frequency of b in the top N snippets of query a, denoted as $freq(b@a)$. Similarly, the backward process counts the frequency of a in the top snippets of query b, denoted as $freq(a@b)$. The *CODC* association score is defined as follows:

$$S_{CODC}(q_f, q_e) = \begin{cases} 0, \text{ if } freq(q_e@q_f) \cdot freq(q_f@q_e) = 0; \\ \exp\left\{\log_{10}\left[\frac{freq(q_e@q_f)}{freq(q_f)} \times \frac{freq(q_f@q_e)}{freq(q_e)}\right]^{\epsilon}\right\}, \text{ otherwise} \end{cases} \quad (3.10)$$

Note that a $CODC$ value ranges between 0 and 1. In one extreme case where $freq(q_e@q_f) = 0$ or $freq(q_f@q_e) = 0$, q_e and q_f have no association; in the other extreme case where $freq(q_e@q_f) = freq(q_f)$ and $freq(q_f@q_e) = freq(q_e)$, they have the strongest association. In our experiment, ϵ is set at 0.15 following [15].

In addition to the frequency feature above, any mined query q_e will be associated with a feature *CODC measure* with $S_{CODC}(q_f, q_e)$ as its value.

3.3.3.4 Monolingual Query Suggestion

For all the candidate queries Q_0 being retrieved using a dictionary (see Section 3.3.3.1), a parallel corpus (see Section 3.3.3.2) and Web mining (see Section 3.3.3.3), the monolingual query suggestion system (see Section 3.3.2) is invoked to produce more related queries in the target language. For each target language query q_e, its monolingual source query $SQ_{ML}(q_e)$ is defined as the query in Q_0 with the highest monolingual similarity with q_e as follows:

$$SQ_{ML}(q_e) = argmax_{q'_e \in Q_0} sim_{ML}(q_e, q'_e) \tag{3.11}$$

The monolingual similarity between q_e and $SQ_{ML}(q_e)$ is used as the value of q_e's *Monolingual Query Suggestion Feature*. For any target query $q \in Q_0$, its *Monolingual Query Suggestion Feature* is set to 1; For any query $q_e \notin Q_0$, its values of *Dictionary-based Translation Score, Bi-Directional Translation Score, Frequency in the Snippet,* and *CODC Measure* are set to be equal to the feature values of $SQ_{ML}(q_e)$.

Following the French query example "pages jaunes" in Figure 3.1, we use Figure 3.2 to illustrate how the CLQS candidate set Q_0 can be replenished by monolingual query suggestions of the candidates available and how their feature values can be set. Suppose Q_0 is initially constructed as shown in the left hand side of Figure 3.1. As we can see from Figure 3.2, the query "white page search" is added to Q_0, and its monolingual query suggestion feature value is set to 0.964, which is the highest

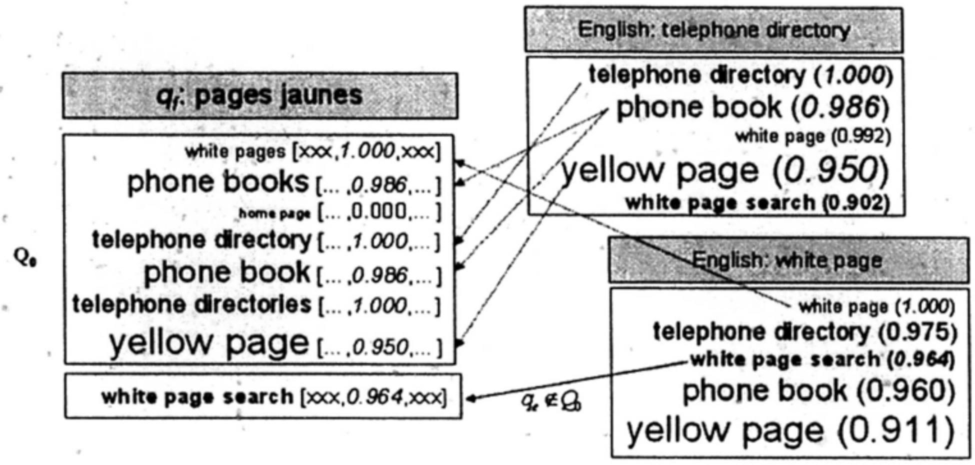

Figure 3.2. An illustration on how the CLQS candidate set Q_0 of French query "pages jaunes" can be updated or replenished by the monolingual query suggestions of the candidates "telephone directory" and "white page". Note that the queries are normalized, and plurals and non-plurals are of no difference.

among with its monolingual source query "white page", and the other feature values of "white page search" are set as the same values as those of "white page".

3.3.4 Learning Cross-lingual Query Similarity Measure

In summary, four categories of features are used to learn the cross-lingual query similarity. SVM regression algorithm [89] is adopted to learn the weights in Equation 3.2. In this study, LibSVM[6] toolkit [13] is employed for the regression training.

In the prediction stage, the candidate queries are ranked using the cross-lingual query similarity score computed using $sim_{CL}(q_f, q_e) = \mathbf{w} \cdot \phi(\mathbf{f}(q_f, q_e))$, and the queries with similarity score lower than a threshold will be regarded as non-relevant. The threshold is learned using a development dataset by fitting MLQS's output. More specifically, we first divided the CLQS candidates into two categories: relevant if a CLQS is in the set of MLQS and non-relevant otherwise. A binary classification

[6]http://www.csie.ntu.edu.tw/~cjlin/libsvm/

model was then trained. The relevancy threshold on the predicted cross-lingual query similarity is determined as the decision boundary of the classifier.

3.4 CLIR Based on CLQS

In Section 3.3, we presented a discriminative model for cross-lingual query suggestion. For benchmarking purpose, we compare the effectiveness of CLQS with query translation for CLIR tasks. The resulting good performance of CLIR presumably corresponds to the high quality of the suggested queries.

Given a source query q_f, a set of relevant queries $\{q_e\}$ in the target language are recommended using the cross-lingual query suggestion system. The suggested queries in $\{q_e\}$ are concatenated to form a long query to retrieve documents. The advantage of this method over the retrieve-then-combine approach is that one can naturally regard the suggested queries as the user's information need as a whole. This resembles the way of how query expansion works by considering feedback terms as natural extension of the original query. For retrieval purpose, three different and widely used IR models are employed in our CLIR experiments, namely, BM25 probabilistic model [79], language modeling-based IR model [72, 103] and vector space model [83].

3.5 Experiments and Results

We benchmark the cross-lingual query suggestion system by comparing its effectiveness with monolingual query suggestion. We study the contribution of different information sources, and test their effectiveness in CLIR tasks. The language pairs concerned are French-English and Chinese-English. Such selection is due to the fact that large-scale query logs are readily available for the two language pairs. Moreover, English is considered as correlated with French more strongly than with Chinese. Thus, we can assume that there is stronger correspondence between the input French queries and the English queries in the query log and such correspondence between

Chinese and English queries is weaker. This enables us to study the effect of different language pairs in CLQS-based CLIR.

3.5.1 Data Resources

Note that French-English (Chinese-English) denotes using French (Chinese) as the source language and English as the target language.

3.5.1.1 English Query Log

We used a one-month English query log of *MSN* search engine (now *Live Search*) in year 2005 as the target-language log. The log contained over 7.01 million unique English queries with occurrence frequency more than 10. A monolingual query suggestion system was built based on the method described in Section 3.3.2. For all the French-English and Chinese-English experiments, we used the same English query log for mining CLQS candidates.

3.5.1.2 French-English Data

In addition to the English query log, we obtained a French query log containing over 3 million queries, from which we selected a set of source queries to build a corpus for learning CLQS model. First, we randomly selected 20,000 French queries from the French log to form a query pool, and automatically translated them into English by Google's machine translation tool. We found that 42.17% (8,433) French queries had corresponding translations in the query English log. Among these French-English query pairs, professional translators then manually selected 4,171 pairs of correct translations. Only these selected query pairs were adopted for learning. Among them 70% were used for cross-lingual query similarity training, 10% are used as the development data to determine the relevancy threshold, and 20% are used for testing.

To retrieve the cross-lingual related queries, a built-in-house French-English bilingual lexicon (containing 120,000 unique entries) and the *Europarl* parallel corpus [46] (with about 1 million French-English parallel sentences from the proceedings of the European Parliament) were also used.

In addition to benchmark CLQS as an independent system, the CLQS system was also evaluated as a query "translation" system for CLIR tasks. The goal was to measure the quality of CLQS in terms of its effectiveness for CLIR. TREC-6 CLIR dataset (AP88-90 English newswire, 750MB) and the officially provided 25 short French-English queries pairs (CL1-CL25) [85] were used for benchmarking. This dataset is readily available. The average length of the title queries in the set is 3.3 words long, which matches the Web queries used to train the CLQS model.

3.5.1.3 Chinese-English Data

We obtained a small Chinese query log of the same period of time with 32,730 queries. From that we selected source queries. First, machine translation was applied to translate the queries into English. We found 21.41% (7,008) Chinese queries had corresponding translations in the English query log. We then manually checked these translations and selected 3,767 correct Chinese-English query pairs which were used for CLQS model training (70%), testing (20%) and development (10%).

To retrieve CLQS candidates, we employed a Chinese-English bilingual lexicon containing 940,000 unique entries and the LDC's Hong Kong parallel corpus (*Catalog No.: LDC2004T08*) with about 3 million parallel sentences.

In CLIR experiments, we performed the NTCIR-4's Chinese-English CLIR task [44]. The English documents were three subsets of the test collection including the news of 1998-99 from Mainichi Daily News, Korea Times, and Xinhua News Agency. The number of document were about 240,490. 60 search topics (001-060) were provided with their translations, and the title field of each topic was selected as the query for retrieval. The average length of the Chinese title queries was 4.4 words, a little longer than the TREC-6 queries, but it was still close to the length of Web

Table 3.1. Main data resources employed in our experiments. Both CLQS and CLQS-based CLIR experiments use the CLQS model trained on 70% of the query translation pairs compiled by human experts to generate cross-lingual query suggestions.

	French-English	**Chinese-English**
# queries in target-language log	7.01 million	7.01 million
# translation pairs by expert	4,171	3,767
% of pairs for CLQS training	70% of 4,171 (2,920)	70% of 3,767 (2,637)
% of pairs for CLQS development	10% of 4,171 (417)	10% of 3,767 (377)
% of pairs for CLQS testing	20% of 4,171 (834)	20% of 3,767 (753)
Size of bilingual dictionary	120,000 entries	940,000 entries
Size of parallel corpus	1 million sentences (Europarl corpus)	3 million sentences (LDC HK parallel corpus)
# CLIR query pairs	25 (TREC-6)	60 (NTCIR-4)
CLIR document collection	AP news (1988-90)	Mainichi Daily News, Korea Times, Xinhua News (1998-99)

queries. NTCIR provides two kinds of relevance judgment, i.e., "Relaxed" relevance and "Rigid" relevance. We based our evaluation on the "Rigid" judgment files.

Before translation, a Chinese query must be appropriately segmented into a sequence of meaningful words. *MSRSeg*[24], a state-of-the-art Chinese word segmenter, was used for this purpose. *MSRSeg* provides a pragmatic mathematical framework to unify five sets of fundamental features of word-level Chinese language processing: lexicon word processing, morphological analysis, factoid detection, named entity recognition, and new word identification.

Table 3.1 summarizes the data resources described above.

3.5.2 CLQS Performance

3.5.2.1 Performance Measure

Mean-square-error (*MSE*) was used to measure the regression error and it is defined as follows:

Table 3.2. French-English CLQS performance with different feature settings (DD: dictionary only; DD+PC: dictionary and parallel corpora; DD+PC+Web: dictionary, parallel corpora, and Web mining; DD+PC+Web+MLQS: dictionary, parallel corpora, Web mining and monolingual query suggestion)

Features	**Regression**	**Classification**	
	MSE	Precision	Recall
DD	0.274	0.723	0.098
DD+PC	0.224	0.713	0.125
DD+PC+Web	0.115	0.808	0.192
DD+PC+Web+MLQS	0.174	0.796	0.421

$$MSE = \frac{1}{l}\sum_{i,j}\left[sim_{CL}(q_f^i, q_e^{ij}) - sim_{ML}(T_{q_f^i}, q_e^{ij})\right]^2 \tag{3.12}$$

where i is the index of the i-th source query in the testing data, j is the index of the suggested queries of the i-th query, and and l is the number of cross-lingual query pairs.

A relevancy threshold was learned using the development data (see Section 3.3.4). Only CLQS with similarity value above the threshold was regarded relevant to the input query. In this way, CLQS was evaluated as a classification task using precision (P) and recall (R) which are defined as follows:

$$P = \frac{|S_{CLQS} \cap S_{MLQS}|}{|S_{CLQS}|}, \quad R = \frac{|S_{CLQS} \cap S_{MLQS}|}{|S_{MLQS}|}$$

where S_{CLQS} is the set of relevant queries suggested by CLQS, S_{MLQS} is the set of relevant queries suggested by MLQS (see Section 3.3.2), and $|.|$ indicates the size of a query set.

3.5.2.2 CLQS Performance

The French-English and Chinese-English CLQS results with different feature configurations are shown in Table 3.2 and Table 3.3, respectively.

The baseline system (DD) used a conventional query translation approach, i.e., a bilingual dictionary for co-occurrence-based translation disambiguation. For French-

Table 3.3. Chinese-English CLQS performance with different feature settings

Features	Regression	Classification	
	MSE	Precision	Recall
DD	0.236	0.854	0.149
DD+PC	0.236	0.892	0.212
DD+PC+Web	0.202	0.824	0.261
DD+PC+Web+MLQS	0.166	0.883	0.442

English CLQS in Table 3.2, the baseline system only covered less than 10% of the suggestions made by MLQS (see recall). Using additional features enabled CLQS to generate more relevant queries. The most significant improvement on recall was achieved by exploiting MLQS. The final CLQS system generated 42% of the queries suggested by MLQS. There was no significant change in precision among all the feature combinations. The performance of Chinese-English CLQS in Table 3.3 showed a similar trend as Table 3.2. This indicated that our method could improve recall without loss in accuracy by effectively leveraging different information sources.

The regression performance was improved with additional features and was consistently reflected by the decrease in regression error (i.e., MSE). This was because the CLQS system increasingly enhanced the cross-lingual query similarity estimation by aligning with the monolingual query similarity under the help of additional information sources.

Chinese-English CLQS performed unexpectedly well. Compared to French-English performance, the high recall values of Chinese-English CLQS were likely the result of the large size of bilingual dictionary and parallel corpus.

The trend of recall improvement with additional information sources also reflects the high quality of the query log where target-language queries are suggested. CLQS is constrained by the size and quality of query log. When log quality is low, it can be expected that the translated query terms have no or only few related queries identified from the log. In that case, no improvement on recall can be increasingly achieved even if one tries to incorporate additional sources. Therefore, query log in reasonably good

international terrorism (0.991);	**what is terrorism** (0.943);
counter terrorism (0.920);	**terrorist** (0.911);
terrorist attacks (0.898);	**international terrorist** (0.853);
world terrorism (0.845);	global terrorism (0.833);
transnational terrorism (0.821);	human rights (0.811);
terrorist groups (0. 777);	patterns of global terrorism (0.762);
september 11 (0.734)	

Figure 3.3. An example of CLQS of the French query "terrorisme international", where the queries suggested by MLQS are shown in bold.

quality is important to the success of CLQS. By observing to find such an improving trend of recall, we can determine qualitatively that a given query log is good enough or too noisy otherwise for CLQS applications. Being aware of this issue, we leave the specific studies on query log quality for future work.

In addition to compare CLQS output with the MLQS output, 200 French queries were randomly selected from the pool of 20,000 French queries. They were double-checked to make sure that they were not in the CLQS training corpus. The CLQS system is then used to suggest relevant English queries for them. On average, for each French query, 8.7 English queries were suggested. A total of 1,740 suggested English queries were manually cross-validated by two professional translators. Among the 1,740 suggested queries, 1,407 queries were deemed as relevant to their original counterparts, hence the accuracy was 80.9%. Figure 3.3 shows an example of CLQS of the French query "terrorisme international" ("international terrorism" in English), among which the queries suggested for the English translation "international terrorism" by MLQS are displayed in bold.

We then conducted similar human evaluation as above for 60 Chinese queries. In average, there were 14.8 English queries suggested for each Chinese query by the system. The total number of suggested queries was 885, among which 748 queries were considered relevant. Therefore, the accuracy of Chinese-English CLQS was 84.5%.

nba michael jordan retired (0.988);	**nba michael jordan retirement** (0.987);
michael and jordan and retired (0.980);	**michael jordan retirement ceremonies** (0.911);
jordan michael (0.843);	**michael jordan** (0.817);
nba jordan retirement (0.799);	nba jordan retired (0.799);
life of michael jordan (0.697);	chicago bulls (0.694)

Figure 3.4. An example of CLQS of the Chinese query "NBA麦可乔丹退休", where the queries suggested by MLQS are shown in bold.

Figure 3.4 shows an example of CLQS of the Chinese query "NBA麦可乔丹退休" ("NBA Michael Jordan retirement").

3.5.3 CLIR Performance

CLQS was evaluated for French-English (F2E) and Chinese-English (C2E) CLIR tasks. We conducted F2E and C2E experiments using the TREC-6 and NTCIR-4 CLIR datasets (see Section 3.5.1), respectively.

CLIR was performed using a query translation system followed by a monolingual IR module based on Lemur's toolkit[7]. Three typical retrieval models were studied separately, i.e., BM25 [79], language modeling-based IR (LM) [72, 103], and TFIDF vector space model (TFIDF) [83]. The following three systems were used to perform query translation:

1. CLQS: Our CLQS systems. The F2E and C2E CLQS models were trained on the respective 70% of human expert compiled French-English and Chinese-English query translation pairs (see Section 3.5.1.2 and 3.5.1.3) with all the features (see Section 5.3) configured.

2. For F2E, we used the Moses translation engine [47], a phrase-based SMT system based on the source-channel formalism [67, 48], denoted as "SMT (Moses)". For C2E, we used a built-in-house SMT system [54, 104], denoted as "SMT

[7] `http://www.lemurproject.org/`

Table 3.4. Average precision of French-English CLIR on TREC-6 dataset (Monolingual: monolingual IR system; DT: CLIR based on dictionary translation; SMT (Moses): CLIR based on Moses statistical machine translation engine; CLQS: CLQS-based CLIR). IR models are tuned to nearly their optimal performance – BM25: $k_1 = 1.2$, $b = 0.75$, $k_3 = 7$; LM: language modeling with Jelinek-Mercer (interpolate) smoothing; TFIDF: query term TF weighting method – Raw-TF, document term TF weighting method – log-TF.

CLIR systems	BM25		LM		TFIDF	
	Average Precision	% of monolingual	Average Precision	% of monolingual	Average Precision	% of monolingual
Monolingual	0.2954	100%	0.2844	100%	0.2739	100%
DT	0.2130	72.11%	0.2115	74.37%	0.1958	71.49%
SMT (Moses)	0.2545	86.15%	0.2412	84.81%	0.2448	89.38%
CLQS	0.2916	98.71%	0.2698	94.87%	0.2585	94.38%

(MSRA)", which also adopted a phrase-based translation model. The two systems represented the state-of-the-art SMT tools for French-English and Chinese-English translation, and were trained on the corresponding sets of parallel corpora used by our CLQS systems (i.e., *Europarl* for F2E and *LDC's Hong Kong corpus* for C2E).

3. DT: A dictionary-based query translation system using co-occurrence statistics for translation disambiguation [4, 26] was applied to the query log (see Section 3.3.3.1). Especially for C2E CLIR, we implemented the approach in [105] to automatically extract OOV translations for Chinese queries from the Web, denoted as "DT (Web)". This represented the state-of-the-art Web mining approach for dictionary-based query translation.

The monolingual IR performance using the standard target language queries was also reported as a reference.

3.5.3.1 F2E CLIR

The average precision of the three F2E CLIR and the monolingual IR systems were reported in Table 3.4 using different retrieval models.

Table 3.5. The p-values result from pair-wise significance t-tests for different French-English CLIR systems. The confidence level is set as 95% ($p < 0.05$ are considered statistically significant)

	BM25		LM		TFIDF	
	DT	SMT (Moses)	DT	SMT (Moses)	DT	SMT(Moses)
CLQS	0.018	0.039	0.028	0.042	0.023	0.047

The result on BM25 retrieval showed that using CLQS as a query translation tool outperformed CLIR based on dictionary translation by 36.9% (relative improvement, i.e. $(0.2916 - 0.213)/0.213$), and machine translation by 14.58%. It achieved 98.71% of the monolingual IR performance. Consistent results were obtained using language modeling and TFIDF vector space model for retrieval. Using language-modeling-based retrieval with Jelinek-Mercer (interpolate) smoothing, CLQS outperformed dictionary-based query translation by 27.57% as well as machine translation by 11.86%, and achieved 94.87% of the monolingual IR performance. Using TFIDF vector space model, CLQS outperformed dictionary-based method by 32.02%, as well as machine translation by 5.6%, and achieved 94.38% of monolingual IR performance. This showed consistent advantage of CLQS-based CLIR over the other traditional query translation approaches. We further conducted test for significance (two-tailed pairwise student's t-test) [35] on the results of different approaches. The p-values shown in Table 3.5 suggested that the performance of CLQS-based CLIR was significantly better at 95% confidence level.

The effectiveness of CLQS lies in its ability in suggesting closely related queries other than accurate translations. For example, consider the query CL14 "terrorisme international" ("international terrorism"). Although MT translated the query correctly, CLQS system still achieved higher score by recommending many additional related terms such as "global terrorism", "world terrorism", etc. (see Figure 3.3). For another example, consider the query CL6 "La pollution causée par l'automobile" ("air pollution due to automobile"). The Moses SMT provided the translation "the pollution caused by cars", but the CLQS system enumerated all possible synonyms of "car", and suggested the queries "car pollution", "auto pollution", "automobile pollution". In addition, other related queries such as "global warming" were also suggested, resulting in an analogous effect of query expansion. For the query CL12 "la culture écologique" ("organic farming"), Moses translated it as "ecological culture", which was not the term used in English. Thus it failed to generate the correct translation and to find the relevant documents. Although the correct translation was neither in our French-English dictionary, CLQS system generated "organic farm" as a relevant query due to successful Web mining.

3.5.3.2 F2E CLIR with Pseudo-Relevance Feedback

The above experiments demonstrated the effectiveness of using CLQS to suggest relevant queries for CLIR enhancement. A related research was to adopt query expansion to enhance CLIR effectiveness [3, 64]. Pseudo-relevance feedback (PRF) is widely used to obtain more alternative query expressions from retrieved documents. Practically, our approach aims to obtain similar effects. Thus, we compared the CLQS approach with the conventional query expansion approaches. Following [64], post-translation expansion was performed based on PRF techniques. We first performed CLIR in the same way as before using different retrieval models. We then applied the traditional PRF algorithms corresponding to the different retrieval models to perform post-translation expansion. Table 3.6 shows the corresponding feedback models with respect to different retrieval models.

For BM25 model, we used the method described in [75] to select expansion terms. In our experiments, the top 10 to 200 terms were selected based on RSV (see Table 3.6) from the top 30 feedback documents to expand the original query for the comparison between CLQS and the baseline approaches. For language modeling approach, PRF was done by using a mixture feedback model described in [102]. Unlike the PRF

Table 3.6. The representative relevance feedback formulations corresponding to the three typical retrieval models: BM25, Language-modeling-base retrieval (LM), and TFIDF vector space model (TFIDF).

IR Model	Relevance Feedback Model	Reference
BM25	$RSV_i = w_i \cdot r_i / R \quad (3.13)$ $w_i = \log \frac{(r_i + 0.5)/(R - r_i + 0.5)}{(n_i - r_i + 0.5)/(N - n_i - R - r_i + 0.5)}$ where RSV_i is the Robertson Selection Value (RSV) for term i; w_i is the Robertson-Sparck Jones relevance weight [77] of the term; r_i is the number of relevant document for the query containing the term; R is the total number of relevant documents for the query; n_i is the number of documents in the collection containing the term; N is the number of indexed documents in the collection.	[75]
LM	$\hat{\theta}_{Q'} = (1 - \alpha)\hat{\theta}_Q + \alpha\hat{\theta}_F \quad (3.14)$ $\hat{\theta}_F \propto \log p(F\|\theta) = \sum_i \sum_w c(w; d_i) \log((1 - \lambda)p(w\|\theta) + \lambda p(w\|C))$ where $\hat{\theta}_{Q'}$ is the updated query model based on the original query model $\hat{\theta}_Q$ and feedback model $\hat{\theta}_F$; α is the coefficient controlling the influence of feedback model; F is the set of feedback documents; $p(F\|\theta)$ is a mixture model used to estimate the feedback model; λ is the parameter controlling the influence of background noise when generating a feedback document.	[102]
TFIDF	$Q_1 = Q_0 + \beta \sum_{k=1}^{n_1} \frac{R_k}{n_1} - \gamma \sum_{k=1}^{n_2} \frac{S_k}{n_2} \quad (3.15)$ where Q_1 is the new query vector, Q_0 is the initial query vector, R_k (S_k) is the vector for relevant (non-relevant) document k, n_1 (n_2) is the number of relevant (non-relevant) documents, and β (γ) is the parameter that control the relative contribution of relevant (non-relevant) documents.	[80]

of BM25, the mixture model updates the query's language model instead of query terms using feedback documents. In addition to varying the number of feedback terms (which is the threshold to truncate the feedback model to no more than the given number of terms), we also examined the influence of feedback model by changing the coefficient α which controlled the extent of inclusion of the feedback model. For TFIDF vector space model, we expanded the queries using the traditional Rocchio's algorithm [80] associated with the vector space model (for the reason of "pseudo" feedback, β was set to 1 and γ to 0). Through the above manual tuning, the three PRF approaches were tuned to their best possible performance. CLIR performances with PRF in terms of average precision using different IR models are shown in Figures 3.5–3.8.

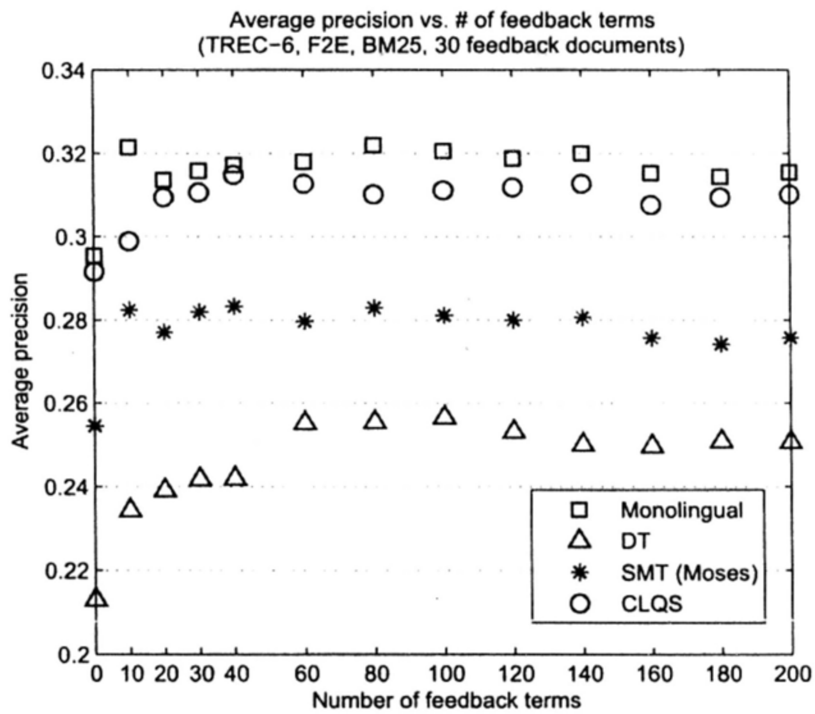

Figure 3.5. Average precision of post-translation expansion using PRF varies with the number of expansion terms on TREC-6 French-English dataset (BM25).

These results showed that the CLQS-based CLIR consistently outperformed the other methods when PRF was incorporated. Especially, even though PRF was not

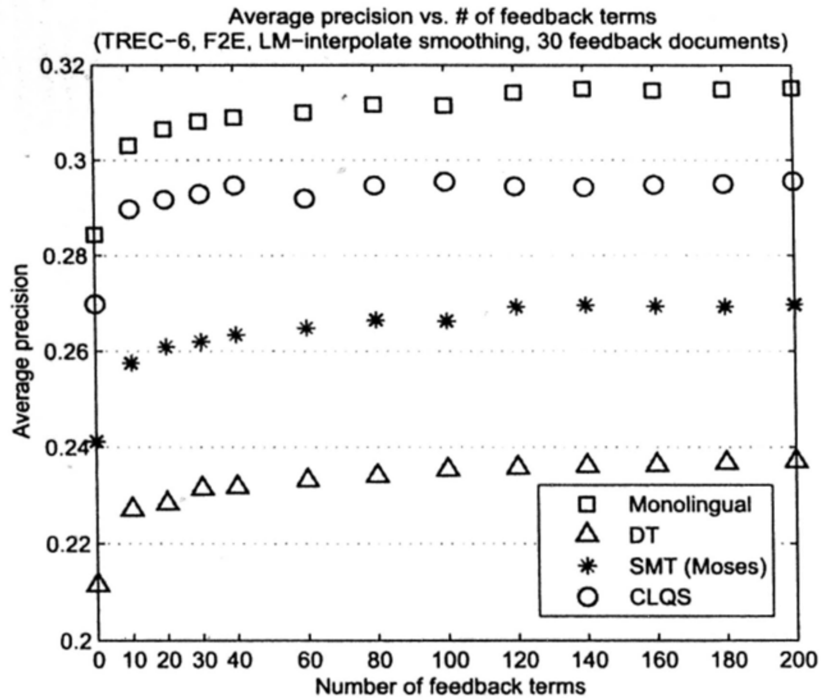

Figure 3.6. Average precision of post-translation expansion using PRF changes with the number of feedback terms on TREC-6 French-English dataset (LM with interpolate smoothing, $\alpha = 0.5$, $\lambda = 0.7$).

added to CLQS-based CLIR (i.e., with zero feedback term), it still performed better than the other two translation approaches plus PRF (with 10+ feedback terms) when using BM25 (see Figure 3.5) and language modeling (see Figure 3.6). In this regard, however, the performance gain was not shown as significant by t-test. We then conducted t-tests with PRF added to CLQS-based retrieval. We found that CLQS-based CLIR with PRF was significantly better than DT-based CLIR with PRF under all the examined number of feedback terms ($p < 0.05$), and was also significantly better than SMT-based retrieval with PRF in most cases, except for BM25 (see Figure 3.5) using 10 feedback terms ($p = 0.095$) and TFIDF (see Figure 3.8) using less than 60 feedback terms (p varies from 0.112 to 0.073).

The results indicated the higher effectiveness of CLQS in related query identification by leveraging a wide range of resources. Post-translation expansion was capable of improving CLQS-based CLIR. This is due to the fact that CLQS and PRF leverage

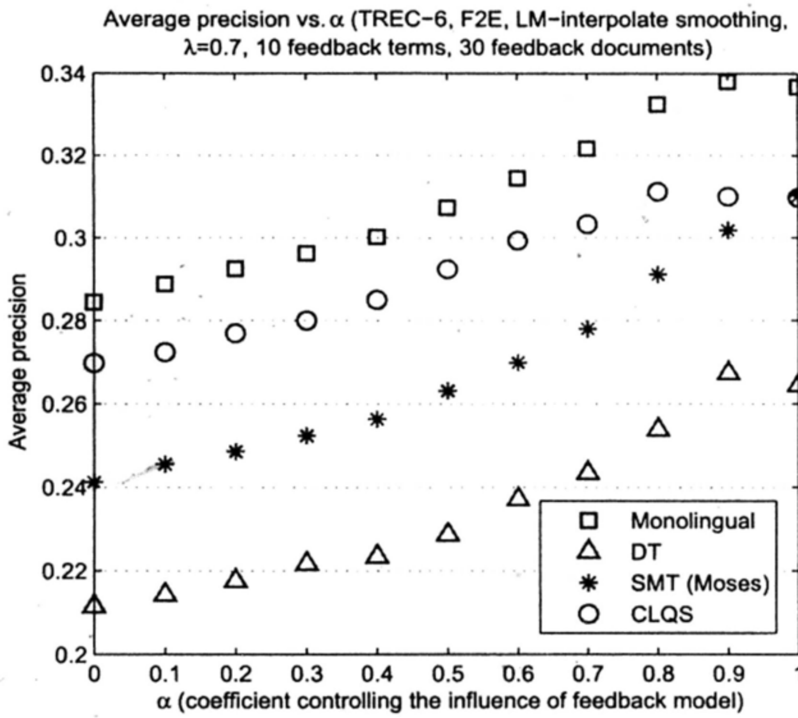

Figure 3.7. Average precision of post-translation expansion using PRF changes with the feedback coefficient α on TREC-6 French-English dataset (LM with interpolate smoothing).

different categories of resources, and both approaches can be complementary. However, the t-test showed that CLQS-based CLIR with PRF was not significantly better than using CLQS alone, and was not always significantly better than other query translation approaches plus PRF especially when only a small number of feedback terms were involved. This may reflect that the related query terms suggested by CLQS from the query log overlapped with the feedback terms from the retrieved documents, and other approaches did not. Thus, introducing a small number of feedback terms was not as helpful to CLQS-based retrieval as to the CLIR based on other query translation approaches. On the other hand, because the queries suggested by CLQS were closely related to the original query, the concatenated long query updated by PRF tended to be more robust to the noise introduced by the feedback process than other query translation approaches. This effect can be observed when the number of

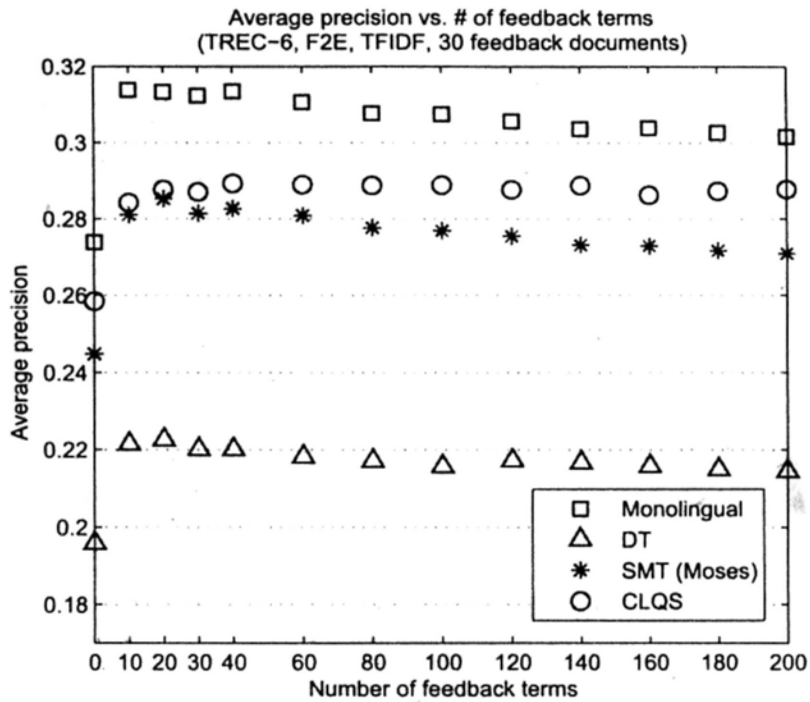

Figure 3.8. Average precision of post-translation expansion using PRF changes with the number of expansion terms on TREC-6 French-English dataset (TFIDF vector space model).

feedback terms increases and no drop in average precision was seen for CLQS (see Figure 3.5 and 3.8).

Using language-modeling-based IR (see Figure 3.6), CLQS was significantly better than other translation approaches regardless of the number of feedback terms. Note that the number of feedback terms in language modeling approach was used to truncate the feedback model (see Equation 3.14) to no more than the given length instead of the number of terms to add to the original query. It seemed that interpolating query model with feedback model improved the effectiveness of CLQS-based CLIR and other query translation approaches in a similar extent give the same truncating threshold. We leave the reason to future study. In addition, average precision stopped increasing for all approaches after certain number of feedback terms were used. This is because the feedback model was truncated when the sum of the probability of the included words reached the default threshold of 1.

Table 3.7. Average precision of Chinese-English CLIR (Rigid test) on NTCIR-4 dataset (Monolingual: monolingual IR system; DT: CLIR based on dictionary translation; DT (Web): CLIR based on dictionary translation with OOV query translations mined from Web; SMT (MSRA): CLIR based on MSRA statistical machine translation engine; CLQS: CLQS-based CLIR). IR models are tuned to nearly their best performance – BM25: $k_1 = 1.2$, $b = 0.75$, $k_3 = 7$; LM: language modeling with Jelinek-Mercer (interpolate) smoothing; TFIDF: query term TF weighting method – Raw-TF, document term TF weighting method – log-TF.

CLIR systems	BM25		LM		TFIDF	
	Average Precision	% of monolingual	Average Precision	% of monolingual	Average Precision	% of monolingual
Monolingual	0.1857	100%	0.1729	100%	0.1733	100%
DT	0.1416	76.25%	0.1302	75.30%	0.1314	75.82%
DT (Web)	0.1564	84.22%	0.1448	83.75%	0.1453	83.84%
SMT (MSRA)	0.1545	83.20%	0.1438	83.17%	0.1389	80.15%
CLQS	0.1720	92.62%	0.1680	97.17%	0.1652	95.33%

We also found that CLQS-based CLIR needed not to rely on a feedback model heavily to boost retrieval performance. This was reflected in Figure 3.7 where the performance of CLQS-based CLIR began to decrease when the influence factor of feedback model got to $alpha = 0.9$. This implies that PRF was less useful to CLQS-based CLIR than to other query translation approaches since certain amount of performance gain was due to the suggested queries themselves.

3.5.3.3 C2E CLIR

The average precisions of the four C2E CLIR and the monolingual IR systems are reported in Table 3.7 in terms of different retrieval models.

Consistent with the F2E CLIR (see Section 3.5.3.1), the higher effectiveness of C2E CLIR based on CLQS shed more light on the advantage of CLQS over the other traditional query translation approaches. When using BM25, CLQS-based CLIR outperformed dictionary-based query translation by 21.47%, dictionary method with OOV translation mining by 9.97%, as well as SMT-based query translation by 11.33%; and achieved 92.62% of the monolingual IR performance. When using language mod-

Table 3.8. The p-values result from pairwise significance t-tests for different Chinese-English CLIR systems. The confidence level is set as 95% ($p < 0.05$ are considered statistically significant).

	BM25		LM		TFIDF	
	DT (Web)	SMT (MSRA)	DT (Web)	SMT (MSRA)	DT (Web)	SMT (MSRA)
CLQS	0.012	0.027	0.0014	0.0006	0.0004	0.0013

eling, CLQS-based CLIR outperformed dictionary-based query translation by 29.03%, dictionary-based query translation plus OOV translation mining by 16.02%, as well as SMT-based query translation by 16.83%; and achieved 97.17% of the monolingual IR performance. When using TFIDF vector space model, CLQS-based CLIR outperformed dictionary-based method by 25.72%, dictionary-based method with OOV translation mining by 13.7%, as well as SMT-based query translation by 18.93%; and achieved 95.33% of monolingual IR performance.

In addition, dictionary-based query translation performed better than machine translation when the OOV translations mined from the Web were added to the dictionary. The machine translation method, however, was constrained by the coverage of the parallel corpus, and could not deal with OOV translations effectively. CLQS leveraged different resources including Web mining of OOV translations to find relevant queries from query log, and covered more relevant information than accurate query translation did. The t-test results shown in Table 3.8 demonstrated that the high effectiveness of CLQS-based CLIR was statistically significant.

For more illustrations, we show some examples from NTCIR-4's query set. For query 005 "戴奥辛 人体 影响 威胁" ("dioxin human body effect threat") where "戴奥辛" ("dioxin") is an OOV term. Both DT and SMT (MSRA) did not correctly translate "戴奥辛" as "dioxin"; but both DT (Web) and CLQS did as they identified the translation pair from the Web corpora. CLQS further suggested related queries in addition to the translated query, such as "how drugs affect the body", "estimated human body burdens dioxin-like chemicals", and "food chain", etc. For query 030 "动

物复制技术" (animal cloning technique), all the methods, except CLQS, did not generate queries with the term "clone" because "clone" was neither a translation entry of "复制" ("reproduction") in our bilingual resources, nor did they co-occur frequently on the Web (what co-occurs more often is "克隆" and "clone"). CLQS correctly suggested "animal cloning technology" as it had a high similarity with "animal reproduction technology clone" in the query log, and MLQS successfully retrieved it from the query log by using "animal reproduction technology", the exact translation of the original query.

3.5.3.4 C2E CLIR with Pseudo-Relevance Feedback

Under similar settings (see Section 3.5.3.2), we compared the average precisions of these different C2E CLIR systems with PRF added. The results are shown in Figures 3.9–3.12.

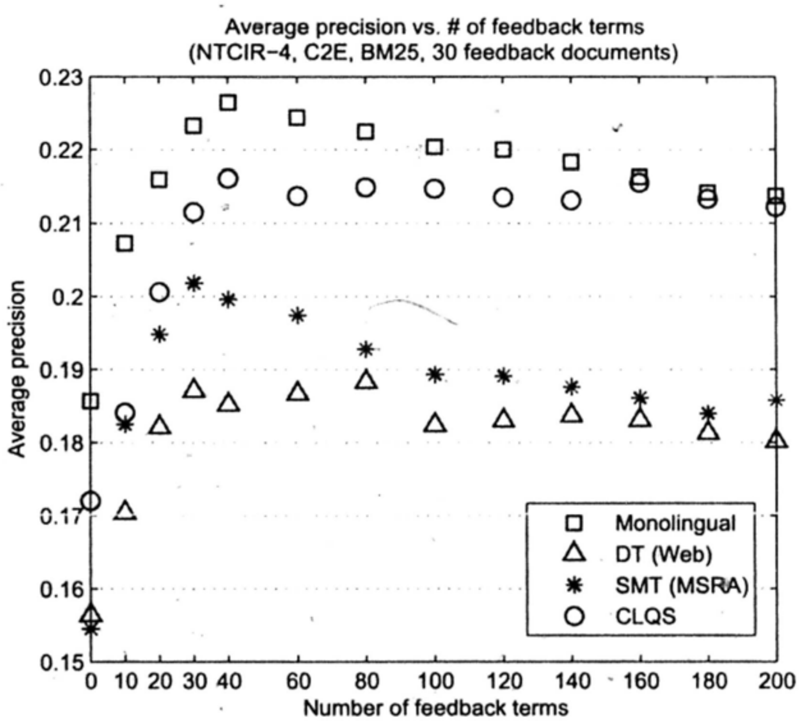

Figure 3.9. Average precision of post-translation expansion using PRF changes with the number of expansion terms on NTCIR-4 Chinese-English (rigid test) dataset (BM25).

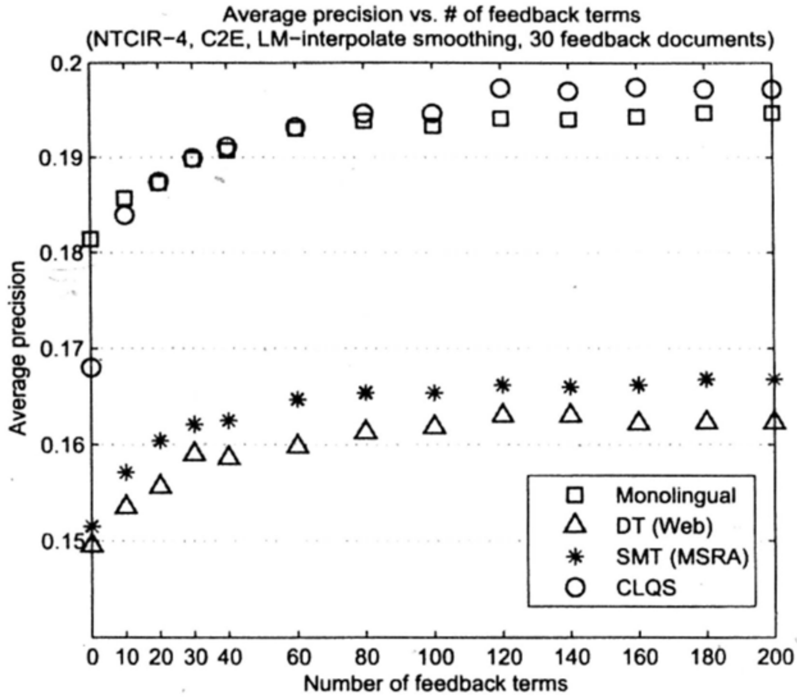

Figure 3.10. Average precision of post-translation expansion using PRF changes with the number of expansion terms on NTCIR-4 Chinese-English (rigid test) dataset (LM with interpolate smoothing, $\alpha = 0.5$, $\lambda = 0.7$).

The results demonstrated that when PRF was performed CLQS-based CLIR showed consistent advantage over the other approaches for a different language pair. In particular, when PRF was not used in CLQS-based CLIR (i.e., with zero feedback term), it still outperformed other query translation approaches plus PRF (with 10+ feedback terms) for language modeling (see Figure 3.10) and TFIDF (see Figure 3.12) except for BM25 (see Figure 3.9). Similarly, t-tests did not show significant performance gain in this regard, but when adding PRF on all retrieval models, CLQS-based CLIR performed significantly better than DT (Web) using any number of feedback terms, and also significantly better than SMT (MSRA) in most cases (p varied from 0.012 to 0.035), except for BM25 using less than 40 feedback terms.

Different from F2E, a t-test between CLQS-based CLIR with and without PRF showed that PRF was not only useful to the CLQS-based approach, but also performed significantly better provided that the appropriate number of feedback terms

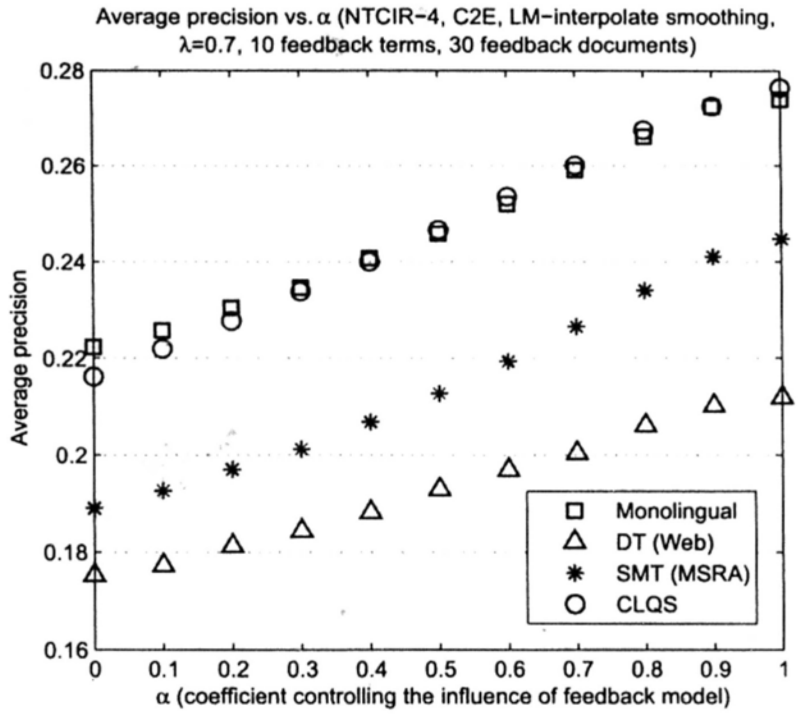

Figure 3.11. Average precision of post-translation expansion using PRF changes with the feedback coefficient on NTCIR-4 Chinese-English (rigid test) dataset (LM with interpolate smoothing).

was used. For example, when more than 20 terms were introduced in the case of BM25 with PRF, the average precision was significantly higher than that of CLQS alone ($p < 0.003$). Such significant improvement was also observed in language modeling as well as in TFIDF with more than 10 feedback terms. This was because C2E CLQS, although effective, could not suggest closely related queries as effectively as its F2E counterpart. Unlike French queries, the Chinese queries were less strongly corresponded to the queries in the English query log due to the wider linguistic gap and the less common search interests of users between the two locales. Thus it was generally harder to find the correspondences of a Chinese query from the English query log than in the F2E case. This observation was reflected by the estimated proportions of Chinese and French queries having corresponding translations in the English query log, i.e., 21.41% vs. 42.17% (see Section 3.5.1.2 and 3.5.1.3). Therefore, the role of

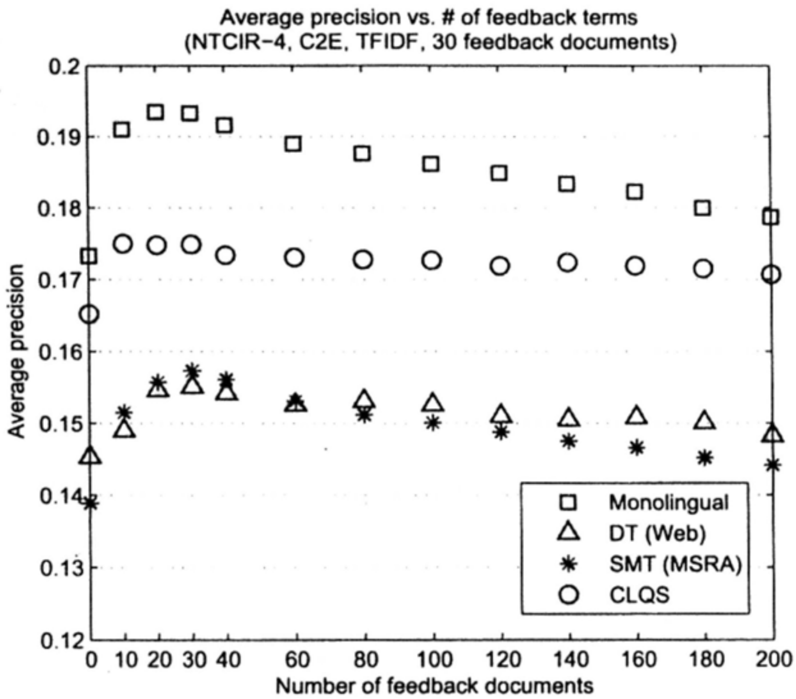

Figure 3.12. Average precision of post-translation expansion using PRF changes with the number of expansion terms on NTCIR-4 Chinese-English (rigid test) dataset (TFIDF vector space model).

PRF was more important in C2E than in F2E for improving CLIR effectiveness. Our conjecture was proven by the results shown in Figure 3.11 (compared to Figure 3.7), where the performance of CLQS-based CLIR increased monotonically with the increasing involvement of the feedback model. This implied that the performance gain increasingly comes from the complementary effect of PRF.

We also noticed that the average precisions dropped remarkably after reaching the peeks with around 30-40 feedback terms for BM25 and TFIDF models (see Figure 3.9 and 3.12). The trend of drop was more evident than the case of F2E. This was due to the errors made during segmenting Chinese texts, which further resulted in a lot more noisy feedback terms in the expansion if the number of feedback terms used was large. It seemed that language-modeling-based retrieval was more robust to this kind of noise (see Figure 3.10). We tried to explain this distinct observation by the factors used to truncate the feedback model, i.e., the constraints on the number of terms used and the sum of the probability threshold of these terms. Another possible reason was that the Kullback-Leibler divergence was not sensitive to the changes in the query model made by the noisy terms since their probability masses were tiny. The discussion on this problem is beyond our scope, which is left to the future work.

3.6 Chapter Summary

In this chapter, we proposed a new approach to cross-lingual query suggestion by mining relevant queries in different languages from query logs. Compared to query translation, our method can suggest not only better formulated queries in the target language, but also similar queries. The key issue to this approach is to learn a cross-lingual query similarity measure between the original query and the suggestion candidates. We proposed a discriminative model to determine such similarity by exploiting different types of monolingual and bilingual information. The model is trained based on the principle that cross-lingual similarity should best fit the monolingual similarity between one query and the other query's translation.

Our method directly follows the intuition that a query in the source language may have its correspondents the query log of target language. This is a straightforward application based on common search interests of different languages at query formulation level.

CHAPTER 4
MULTILINGUAL AND CROSS-LINGUAL RANKING

In Chapter 3, we presented the use of common search interests discovered from the query logs of different languages for Cross-Lingual Query Suggestion and CLIR applications. In this chapter, we propose to generalize the usefulness of common search interest by concentrating on its application for the relevance ranking of retrieved documents in multilingual as well as cross-language information retrieval.

Ranking for multilingual information retrieval (MLIR) is a task to rank documents of different languages based on their relevancy from the source query's perspective. Existing approaches focused on combining separate relevance scores resulted from the monolingual retrieval models of different languages, and did not learn a multilingual ranking function directly. We approach Web MLIR ranking problem using the learning-to-rank framework. Besides adopting popular learning-to-rank algorithms to MLIR, a joint ranking model is proposed to exploit the correlations among documents and induce the joint relevance probability for all retrieved documents irrespective of their origin. Using this method, the relevant documents of one language can be leveraged to improve the relevance estimation for documents of different languages. A probabilistic graphical model is trained for the joint relevance estimation. Especially, a hidden layer of nodes is introduced to represent the salient topics among the retrieved documents. The ranks of the relevant documents and the topics are determined collaboratively during the course of the model approaching to its thermal equilibrium. Furthermore, the model parameters are trained under two settings: (1) optimizing the accuracy of identifying relevant documents; (2) directly optimizing information retrieval evaluation measures, such as mean average precision. It is straightforward that CLIR ranking also can be enhanced by naturally removing the retrieved source-language documents from the output.

The materials presented in this chapter are partially based on our work published in [30].

4.1 Introduction

With the growing volume of information in different languages on the Web, searching across multiple languages is becoming increasingly important. MLIR for Web pages however remains challenging because the documents in different languages have to be compared and merged appropriately. It is hard to estimate cross-lingual relevance due to the information loss from query translation.

Recently, machine learning approaches for ranking in information retrieval, known as learning to rank, have received extensive attention [10, 22, 33, 101]. The learning task is to optimize a ranking function given the data consisting of queries, the retrieved documents and their relevance judgments made by human. Given a new query, the learned function is used to predict the order of the retrieved documents.

However, there is little research to adapt the state-of-the-art ranking algorithms for MLIR. Existing techniques usually combine query translation and monolingual retrieval to derive a relevance score for each document. The relevance scores from different settings are then normalized and made comparable for merging and final ranking [55, 84, 87]. Such approaches do not directly incorporate any feature related to MLIR relevancy. Hence they do not work well for multilingual Web search, where a large number of relevance features can be utilized.

Multilingual learning to rank method aims to optimize a unique ranking function for documents of different languages. This can be done intuitively by representing documents in a unified feature space followed by monolingual ranking. Nevertheless, information loss and misinterpretation in query translation make relevance features between query and individual documents (especially in the target language) inaccurate, rendering multilingual ranking a more difficult problem. For example, English query about the movie "The Matrix" is literally translated to "矩阵" in Chinese, which may result in a number of irrelevant Chinese documents about mathematical matrix. As a consequence, the relevance between the source language query and the documents is distorted, and the retrieved Chinese documents containing keywords "矩阵重装上阵" (The Matrix Reloaded) and "矩阵革命" (The Matrix Revolutions) may be ordered behind the irrelevant ones.

In this work, we propose to leverage on the relevance among candidate documents to improve MLIR ranking. Since similar documents usually share similar ranks, cross-lingual relevant documents can be utilized to enhance relevance estimation for documents of different languages, and hence complement the inaccuracies caused by query translation errors. Given a set of candidate documents, multilingual clustering is performed to identify their salient topics. A probabilistic graphical model, referred to as *Boltzmann machine* (BM) [1, 45], is then used to estimate the *joint relevance probability of all documents* based on both of query-document relevance and relevance between the documents and topics. Furthermore, we train our model by two means: (1) optimizing the accuracy of identifying relevant documents; and (2) directly optimizing IR evaluation measures. Our model can also be applied to CLIR task directly. This can be achieved in a straightforward way by simply removing the source language results from the output. We show significant advantages of our method for both a CLIR task of TREC and a multilingual Web search task under English-to-Chinese retrieval settings.

This chapter is organized as follows: Section 4.2 presents the related work of MLIR ranking; Section 4.3 presents MLIR ranking based on learning-to-rank framework;

Section 4.4 describes our joint ranking model; Section 4.5 discusses the experiments and results; finally, we summarize the chapter in Section 4.6.

4.2 Related Work

MLIR is a task to retrieve relevant documents in multiple languages. Typically, the queries are first translated using a bilingual dictionary, machine translation software or a parallel corpus. The translated queries are then submitted to monolingual retrievals. A re-ranking then takes place to merge different ranked lists from IR in different languages into a single reasonable list.

The Round-Robin approach [95] takes one document in turn from each individual list to build the final list. Score combination approach was proposed by Fox and Shaw [21], where the results from different lists are combined and ranked by the retrieval status value of each document. For the same document, the combined score is calculated using different combination schemes, such as the CombMin, CombMax, CombSUM, CombANZ and CombMNZ, based on its scores in different ranked lists. The combination can be done using the raw score, assuming that the relevance scores of different lists are directly comparable. However, different retrieval results can generate quite different ranges of relevance values, and a normalization method should be first applied to each result before merging [53].

Most existing work focuses on how to combine retrieved items with incomparable scores associated with each result list. Different normalization strategies were proposed by Savoy and Berger [84], such as Norm Max, Norm RSV, and Z-score. They also introduced an advanced method using logistic regression to predict the probability of binary relevancy (relevant or not) for a document according to the logarithm of its rank and the original retrieval status value (RSV). The relevance probability is formulated as a logistic function below:

$$Pr\left[rel(d)|rank(d), RSV(d)\right] = \frac{e^{\alpha+\beta_1\cdot\ln(rank(d))+\beta_2\cdot RSV(d)}}{1+e^{\alpha+\beta_1\cdot\ln(rank(d))+\beta_2\cdot RSV(d)}} \tag{4.1}$$

Based on the relevance probability, they sorted the documents from different lists. Training data are required to estimate the underlying parameters α, β_1 and β_2.

One particular shortcoming of previous combination methods is that they treat the votes from multiple systems with equal weights. Si and Callan [87] proposed a learning approach for score combination. The idea is to develop a better score normalization method and the weights of systems with machine learning over a set of training data. The score function is defined as a weighted combination of scores from M individual ranked lists:

$$score(d) = \frac{1}{M}\sum_{m=1}^{M} w_m \cdot score_m(d)^{r_m}, \tag{4.2}$$

where M is the number of ranked lists, and w_m and r_m are the weight of the vote and the exponential normalization factor for the m-th ranked lists respectively. To derive the parameters, Mean Average Precision (MAP) [74, 78] criterion is used to optimize the accuracy for the training queries as below:

$$(\vec{w},\vec{r})^* = \underset{(\vec{w},\vec{r})}{\operatorname{argmax}}\left[\log(MAP) - \sum_{m=1}^{M}\frac{(w_m-1)^2}{2*a} - \sum_{m=1}^{M}\frac{(r_m-1)^2}{2*b}\right] \tag{4.3}$$

where two regularization coefficients a and b are introduced to avoid overfitting.

Although the work by [84, 87] involved learning, they primarily focused on adjusting the scores of documents from different monolingual result lists and neglected direct modeling of different types of features for measuring MLIR relevancy. Different from the score combination approaches, Tsai et al. [94] directly applied learning-to-rank method to MLIR ranking. They linearly combined a merge model and a BM25 retrieval model:

$$score(d) = (1-\lambda) * M_t(d) + \lambda * bm25(d) \tag{4.4}$$

where λ was the interpolating weight of BM25, and the merge model M_t was a ranking function combining t number of selected weak learners (the ranking function after t-th iteration) to minimize the pairwise fidelity loss [93], i.e., $M_t(d) = \sum_t \alpha_t m_t(d)$, where α_t was the weight of weak learner $m_t(.)$. Each weak learner corresponded to the features selected at three levels: document level, translation level and query level. This method was the first attempt of using learning-to-rank algorithm for MLIR ranking task. But the feature set did not take into account of the relevancy between query and documents. In contrast, our approach focuses on a joint ranking model which learns the ranking function by leveraging the correlation of relevancy (or commonality) among documents of different languages in addition to the query-document relevancy features.

4.3 Learning for Multilingual Ranking

The learning framework for MLIR ranking aims to learn a unique ranking function to estimate comparable scores for documents in different languages. An important step is to design a unified multilingual feature space for the documents. Based on these features, existing monolingual learning-to-rank algorithms can be applied to MLIR ranking. We will first introduce the framework and algorithms of learning to rank, and then give details about constructing the multilingual feature space.

4.3.1 Learning to Rank Framework

The general learning framework for ranking described in [56] is illustrated in Figure 4.1.

Suppose that each query $q \in Q$ (Q is a given query set) is associated with a list of retrieved documents $D_q = \{d_i\}$ and their relevance labels $L_q = \{l_i\}$, where l_i is the rank label of d_i and may take one of the m rank levels in the set $R = \{r_1, r_2, \ldots, r_m\}$,

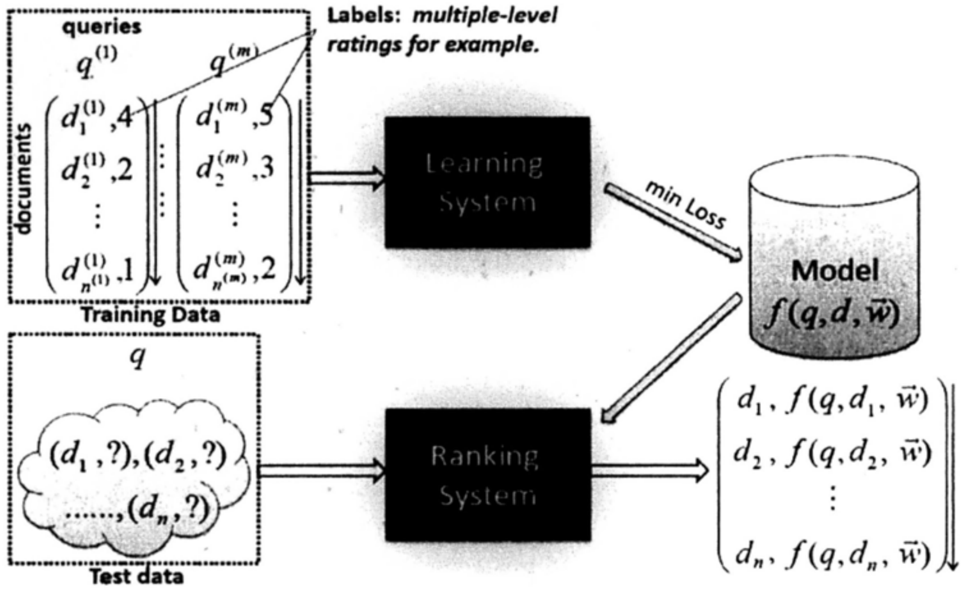

Figure 4.1. The general framework of learning to rank for information retrieval [56].

where $r_1 \succ r_2 \succ \ldots \succ r_m$, and $\succ$ denotes the order relation. The training corpus is then represented as $\{q \in Q | D_q, L_q\}$.

For each query-document pair (q, d_i), we denote the feature space as $\Phi : \vec{\phi}(q, d_i) = [\phi_k(q, d_i)]_{k=1}^{K}$, where $\vec{\phi}$ is the feature vector consisting of K number of relevancy features with respect to (q, d_i), and ϕ_k is one of the feature functions. The goal is to learn a scoring function (or ranking function) $f(q, d_i, \vec{w}) : \Phi \rightarrow \mathbb{R}$ ($\mathbb{R}$ is the real value space) to assign a relevance score for the feature vector of each retrieved document, where $\vec{w}$ is the vector of feature weights. Specifically, a permutation of integers $\pi(q, D_q, f)$ is introduced to denote the order among the documents in D_q ranked by f, and each integer $\pi(d_i)$ refers to the position of d_i in the result list. The objective of ranking is to search for an optimal scoring function:

$$\hat{f} = \operatorname*{argmin}_{f} \sum_{q} E(\pi(q, D_q, f), L_q), \tag{4.5}$$

which minimizes a loss function E that represents the disagreement between $\pi(q, D_q, f)$ and the desirable rank order given by L_q over all queries. The ranking function and loss function may take various forms in different ranking algorithms.

4.3.2 Learning to Rank Algorithms

Based on their input and output spaces, three categories of learning-to-rank approaches have been proposed in recent years, i.e., pointwise approach, pairwise approach and listwise approach.

In pointwise approach, standard probabilistic classification (e.g., Support Vector Classifier) and metric regression (e.g., Support Vector Regression) are typically used for ranking by predicting the rank labels or scores of individual documents. Most of the popular ranking models like Ranking SVM (large-margin ordinal regression) [33], RankBoost [22], RankNet [10], etc., aim to optimize pairwise loss based on order preference and classify the relevance order between a pair of documents, thus falling into the pairwise approach. More recently, listwise approach is proposed to consider the entire group of documents associated with the same query in the input space of the algorithms. Compared to the pointwise and pairwise approaches, the advantage of the listwise approach lies in that its loss function can naturally take into account of the positions of the documents in the ranked list of each query. Since most IR evaluation measures, such as MAP [74, 78] and Normalized Discounted Cumulative Gain (NDCG) [38], are are position-based and used at query level, listwise algorithms can directly optimize these measures.

Although the listwise approach with direct optimization methods is advantageous, the task of optimization is non-trivial as the IR evaluation measures are commonly position-based, and thus non-continuous and non-differentiable. To overcome this predicament, different attempts have been made, including (1) to optimize a continuous and differentiable approximation of an evaluation measure, such as SoftRank [92];

(2) alternatively to optimize a continuous and differentiable (even convex) bound of the evaluation measure, such as SVM-MAP [101]; and (3) to use techniques like Boosting or genetic algorithms which can optimize complex objects, such as AdaRank [100]. A comprehensive survey of these methods are given in [56].

4.3.3 Multilingual Ranking Features

In monolingual ranking, all documents are represented by vectors of features, which reflect the relevance of the documents to the query in a wide range of information retrieval measures. Since it is almost impossible to use only a few factors to satisfy complex information needs of Web users, the capability of combining a large number of features is highly demanding for search engines to easily incorporate the output of any new progress on retrieval model as one dimension of the features.

For each Web page of a given query, *query-dependent features* can be extracted from four different sources of information: the anchor text, the URL, and the title and body of the page. Also, *query-independent features* can be extracted considering the link structure of Web pages, such as PageRank and HITS. [57] presented details of these popular features. The task of learning is to derive the optimal way of combining these features.

Monolingual search engines of different languages may adopt different sets of features, rendering a uniform feature space for multilingual ranking difficult. Some features available in one language may be missing in another language and the feature values are hardly comparable across different languages. Therefore, a key step to learn for multilingual ranking is to design a unified multilingual feature space for documents in different languages. With this unified features space, existing monolingual learning-to-rank algorithms can then be applied for multilingual ranking.

Without loss of generality, suppose we would like to rank documents retrieved from an English and a Chinese corpora. To deal with the problems of missing values and incomparable values of features from different languages, a unified bilingual feature space can be constructed based on the following rules:

1. Any feature values will be normalized using Min-Max algorithm, and then move the value ranges to start from 1;

2. Any query-independent features being used in both corpora will be kept in the new feature space;

3. Any query-independent features being used in only one of the corpora will be kept in the new features space, and the corresponding features of documents from the other language will be assigned a value of 0;

4. Any query-dependent feature ϕ used in the English corpus will be given a new notation ϕ_e, and this feature on the Chinese documents will be assigned a value of 0;

5. Any query-dependent feature ϕ used in the Chinese corpus will be given a new notation ϕ_c, and this feature on the English documents will be assigned a value of 0. Note that the value of ϕ_c for a Chinese document is estimated based on the Chinese translation of the English query.

Figure 4.2 shows an example illustrating the construction of the unified bilingual feature space based on the rules above. With this unified multilingual feature space, popular learning-to-rank algorithms, such as Ranking SVM and RankNet, can be applied to MLIR ranking.

4.4 Joint Ranking Model for MLIR

Although monolingual ranking algorithms can be applied to MLIR, the information loss caused by query translation can seriously affect ranking effectiveness. To complement query-document relevance, we propose a joint ranking model to exploit

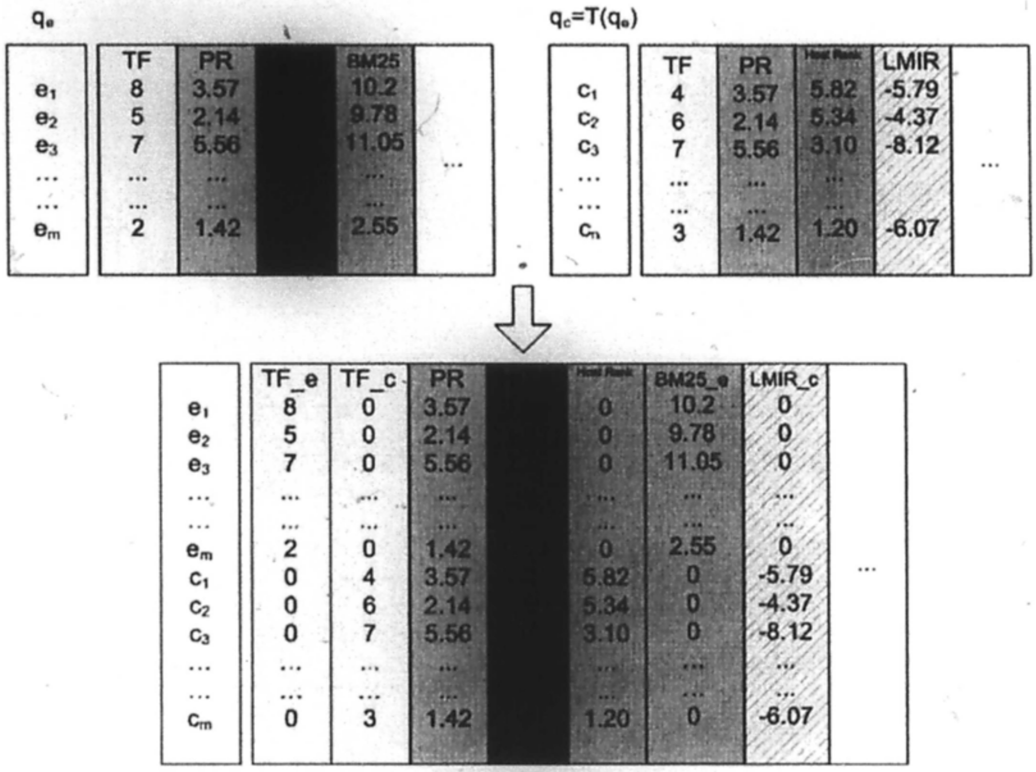

Figure 4.2. An example of constructing the unified multilingual feature space. Note that for ease of reading, the feature values shown underneath are not normalized using rule 1 above so that readers can connect the renewed values with their raw values.

the relationship among documents in different languages. If two documents are bilingually correlated or similar, and one of them is relevant to the query, it is very likely that the other is also relevant. By modeling the similarity, relevant documents in one language may help the relevance estimation of documents in a different language, leading to improve the overall relevance estimation. This can be considered as a variant of pseudo relevance feedback. In our study, *Boltzmann machine* (BM) [1] is used to estimate the joint relevance probability distribution as it is well generalized to model any relationship among objects.

Although joint probabilistic model was not used for IR ranking, similar approaches have been proposed for question answering (QA) and information extraction tasks. For example, in order to exploit the dependency in classifying similar answers, an BM model was used to predict the joint probability of the answers' correctness in the QA task [45]. In [9], Relational Markov Network was applied to enhance protein name extraction from biomedical texts by modeling the dependencies among individual entity candidates. Also, BM was used to model movie rating in collaborative filtering [81]. Note that in previous work the instance pairs were directly linked to represent the interaction between instances. This rendered the size of graph being quadratic to the number of instances, which is formidable to Web search ranking task.

4.4.1 Boltzmann Machine (BM) Learning

BM is an undirected graphical model that makes stochastic predictions about which state values its nodes should take [1]. The global state $\mathbf{s}$ of the graph is represented by a vector $\mathbf{s} = [s_1 s_2 \dots s_n]$, where $s_i = \pm 1$ is the state of the node i and n is the total number of graph nodes. The system's energy under a global state is defined as

$$E(\mathbf{s}) = -\frac{1}{2}\sum_{ij} \theta_{ij} s_i s_j - \sum_i w_i s_i, \tag{4.6}$$

where θ_{ij} is the edge weight between node i and j, w_i is the threshold of node i. Theoretically, after running a stochastic dynamic process for some enough time, the system will reach a state of thermal equilibrium, in which the probability to find the graph in the global state depends only on the states of each node and its neighbors, and follows Boltzmann distribution, i.e., $P(\mathbf{s}) = \frac{1}{Z}\exp(-E(\mathbf{s}))$, where $Z = \sum_{\mathbf{s}} \exp(-E(\mathbf{s}))$ is the normalization function over all possible states.

The training of a machine is to determine the weights and thresholds in such a way that the Boltzmann distribution approximates the target distribution $\tilde{P}(\mathbf{s})$ as closely as possible. The difference between the two distributions is measured by *Kullback-Leibler (K-L) Divergence* [50]: $K(\tilde{P}||P) = \sum_{\mathbf{s}} \tilde{P}(\mathbf{s}) \log \frac{\tilde{P}(\mathbf{s})}{P(\mathbf{s})}$. The objective is to minimize the divergence using gradient descent. The weight updating rules of the following form can be obtained [1]:

$$\Delta\theta_{ij} = \alpha\left(\langle s_i s_j\rangle_{clamped} - \langle s_i s_j\rangle_{free}\right) \tag{4.7}$$

$$\Delta w_i = \alpha(\langle s_i \rangle_{clamped} - \langle s_i \rangle_{free}) \quad (4.8)$$

where α is the learning rate, and $\langle . \rangle_{clamped}$ and $\langle . \rangle_{free}$ denote the expectation values of the node states obtained from the *clamped* and *free-running* stages in training respectively. In clamped stage, states are fixed to the patterns in the training data; in free-running stage, states are changed based on the model's stochastic decision rule. The procedure alternates between the two stages until the model converges.

4.4.2 Joint Relevance Estimation Based on BM

We unambiguously denote the original query and its translation as one query q without differentiating them, i.e., a couple of bilingual queries which reflect a common search interest from users of different languages. Unless mentioned otherwise, q is referred to as a couple of bilingual queries above throughout this chapter, but it is important to note that they are searched independently within their individual language domains.

For each query, one can intuitively represent the retrieved documents as nodes, the correlations between them as edges, and the rank label of each document as node state. Each BM then naturally corresponds to the instances of one query. However, the number of edges is quadratic to the number of documents with this representation. This is unacceptable for Web search where hundreds of candidate documents will be returned for a query. Our idea is to first discover the salient topics using a clustering technique, and the direct document connections are replaced by the edges between the documents and the topics. In particular, only the set of top largest clusters are kept so that the size of the graph's connectivity is linear with the number of documents.

Figure 4.3 shows the representation of a BM for ranking. Correspondingly, we have two types of nodes, i.e., topic nodes and document nodes. Topic nodes represent clusters of retrieved documents. This representation helps reduce complexity. For salient topics, we perform multilingual clustering on the retrieved documents of each

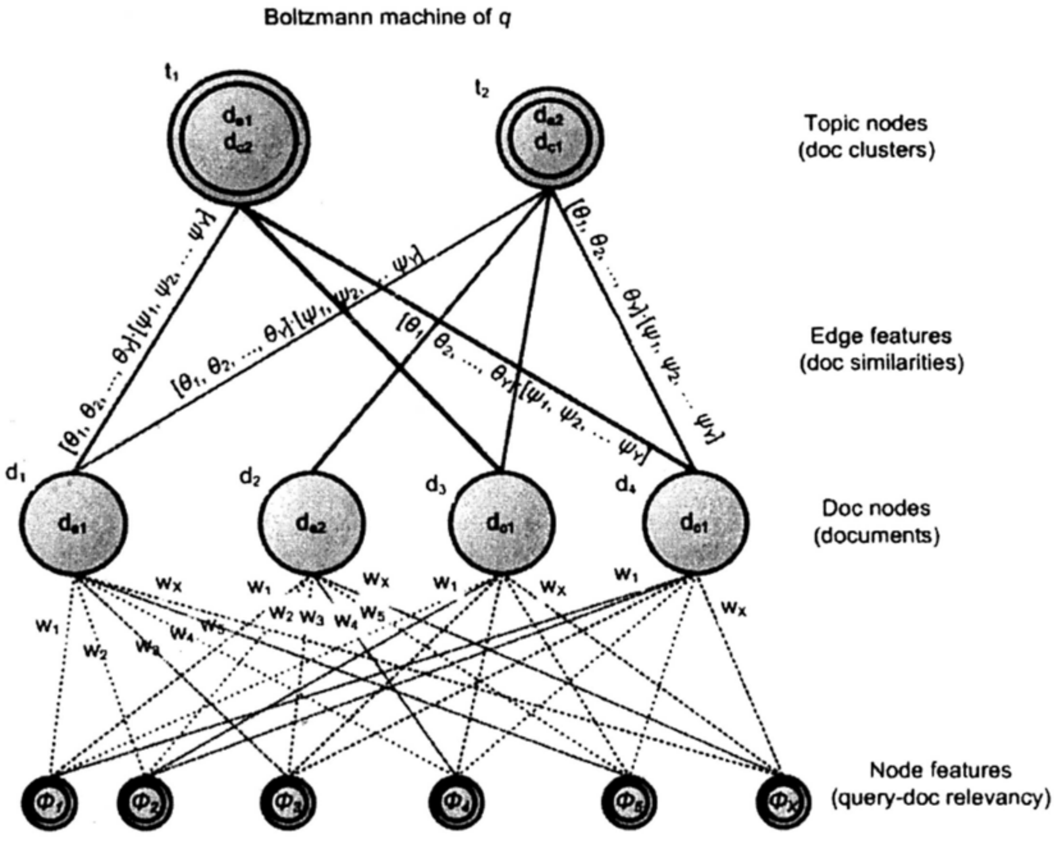

Figure 4.3. Illustration of a Boltzmann machine of a query for MLIR ranking. The top layer contains hidden nodes corresponding to clusters, and the middle layer contains output nodes corresponding to documents. Edges between every document and topic nodes correspond to the correlation between documents and clusters (topics). The bottom layer contains input query-document relevancy features.

query q (see Section 4.4.3). We denote q's salient topic set as $T_q = \{t_j\}$. T_q and D_q correspond to different types of nodes in a graph. Topic nodes are regarded hidden units because their states (rank labels) are not explicitly provided, and document nodes are output units as their rank labels will be the output of ranking. Although a document belongs to only one of the topics, the edges between a document node and every topic node represent the strength of their correlations.

For each query q, suppose n documents are retrieved from different languages and m salient topics are identified in them. We denote $\mathbf{sd}_q = [sd_i]_{i=1}^{n}$ and $\mathbf{st}_q = [st_j]_{j=1}^{m}$ as the state vectors of the document nodes and topic nodes respectively, then the energy of the machine becomes:

$$E(\mathbf{s}, q) = E(\mathbf{sd}_q, \mathbf{st}_q, q) = -\sum_i \vec{w} \cdot \vec{\phi}(q, d_i) sd_i - \frac{1}{2} \sum_{i,j} \vec{\theta} \cdot \vec{\psi}(d_i, t_j) sd_i st_j \tag{4.9}$$

where $\vec{\phi} = [\phi_x(q, d_i)]_{x=1}^{X}$ and $\vec{\psi} = [\psi_y(d_i, t_j)]_{y=1}^{Y}$ are the X-dimension vector of query-document relevancy features on document nodes (i.e., node features) and the Y-dimension vector of document-topic similarities on edges (i.e., edge features) respectively, and $\vec{w}$ and $\vec{\theta}$ are their corresponding weight vectors. Note that one machine corresponds to only one query, and the corresponding weights of node features and edge features are shared across different machines (or queries). The probability of the global state then follows Boltzmann distribution which takes the following form:

$$P(\mathbf{s}, q) = P(\mathbf{sd}_q, \mathbf{st}_q, q) = \frac{1}{Z} \exp\left[-E(\mathbf{sd}_q, \mathbf{st}_q, q)\right] \tag{4.10}$$

where $Z = \sum_{\{sd\}\{st\}} \exp\left[-E(\mathbf{sd}_q, \mathbf{st}_q, q)\right]$ is the normalization constant (see Section 4.4.1).

Although with a hidden layer, note that our joint probabilistic model is different from graphical models for topic discovery, such as Latent Dirichlet Allocation (LDA) [5]. LDA is a hierarchical Bayesian model that views each document as a mixture of various topics. The goal is to estimate the posterior distribution of the hidden topic variables given a document. Different from relevance features in ranking, the topics and documents in LDA are represented as $tfidf$ vectors of words in a collection. When applied in document classification, LDA is used in pre-processing for dimensionality reduction followed by specific classification algorithm. In particular, LDA reduces any document to a fixed set of real-valued features – the posterior Dirichlet parameters associated with the document. Different from LDA, Boltzmann machine is a probabilistic classification model based on joint probability estimation of rank labels using fixed sets of query-document relevance features and inter-document correlations.

4.4.3 Multilingual Clustering for Identifying Salient Topics

For clustering and measuring the relevance among documents, some translation mechanism has to be employed for comparing the similarity of documents in different languages. We use the cross-lingual document similarity measure described in [62] for its simplicity and efficiency. The measure is a cosine-like function with an extension of TF-IDF weights for the cross-lingual case, using a dictionary for keyword translation. The measure is defined as follows:

$$sim(d_1, d_2) = \frac{\sum_{(t_1,t_2)\in T(d_1,d_2)} tf(t_1, d_1)idf(t_1, t_2)tf(t_2, d_2)idf(t_1, t_2)}{\sqrt{Z'}} \tag{4.11}$$

where Z' is given as

$$\begin{aligned} Z' = & \left[\sum_{(t_1,t_2)\in T(d_1,d_2)} (tf(t_1, d_1)idf(t_1, t_2))^2 + \sum_{t_1\in \overline{T}(d_1,d_2)} (tf(t_1, d_1)idf(t_1))^2\right] \times \\ & \left[\sum_{(t_1,t_2)\in T(d_1,d_2)} (tf(t_2, d_2)idf(t_1, t_2))^2 + \sum_{t_2\in \overline{T}(d_2,d_1)} (tf(t_2, d_2)idf(t_2))^2\right] \end{aligned}$$

$T(d_1, d_2)$ denotes the sets of word pairs where t_2 is the translation of t_1, and t_1 (t_2) occurs in document d_1 (d_2). $\overline{T}(d_1, d_2)$ denotes the set of terms in d_1 that have no translation in d_2 ($\overline{T}(d_1, d_2)$ is defined similarly). $idf(t_1, t_2)$ is defined as the extension of the standard IDF for a translation pair (t_1, t_2):

$$idf(t_1, t_2) = \log\left(\frac{n}{df(t_1) + df(t_2)}\right) \tag{4.12}$$

where n denotes the total number of documents in two languages and df is the word's document frequency. In our work, the cross-lingual document similarity is measured as such, and the monolingual similarity is calculated by the classical cosine function [82]. K-means algorithm is used for clustering. We introduce only k largest clusters into the graph as salient topics, where k is chosen empirically ($k = 6$ achieves best results in our case) based on the observation that minor clusters are usually irrelevant to the query.

Equation 4.11 is also used to compute the edge features, i.e., the similarities between documents and salient topics. The edge features for each document-topic pair are defined as 12 similarity values based on the following combinations of three aspects of information: (1) the language — monolingual or cross-lingual similarity depending on the languages of two documents concerned; (2) the field of text — the similarity is computed based on title, body or title+body; and (3) how to do the average for the value — averaging the similarity values with all the documents in the cluster or calculating the similarity between the document and the cluster's centroid.

4.4.4 BM Training as a Classifier

The training is to adjust the weights and thresholds in such a way that for each query the predicted probability of document relevancy, i.e., the model $P(\mathbf{sd}_q, q) = \sum_{\mathbf{st}_q} P(\mathbf{sd}_q, \mathbf{st}_q, q)$, approximates the target distribution $\tilde{P}(\mathbf{sd}_q, q)$ as closely as possible, where

$$\tilde{P}(\mathbf{sd}_q, q) = \begin{cases} 1, & \text{if} \quad \mathbf{sd}_q = L_q; \\ 0, & \text{otherwise} \end{cases}$$

is obtained from the training data. By minimizing the *K-L Divergence*, we obtain the weight update rules

$$\begin{aligned} \Delta w_x &= \alpha \sum_q \left(\left\langle \sum_i sd_i \cdot \phi_x(q, d_i) \right\rangle_{clamped} - \left\langle \sum_i sd_i \cdot \phi_x(q, d_i) \right\rangle_{free} \right) \\ \Delta \theta_y &= \alpha \sum_q \left(\left\langle \sum_{i,j} sd_i \cdot st_j \cdot \psi_y(d_i, t_j) \right\rangle_{clamped} - \left\langle \sum_{i,j} sd_i \cdot st_j \cdot \psi_y(d_i, t_j) \right\rangle_{free} \right) \end{aligned}$$

which bear similar forms as Equations 4.7 and 4.8, and it is noted that the expectation values calculated here are of features and similarities rather than node states.

The training procedure alternates between the clamped and free stages. It is repeated several times with different initial weight values to avoid local optima. Unlike an output unit whose state is fixed to its human label in the clamped stage, the state value of a hidden unit (i.e., a topic) is decided by the model in both stages. Note that the exact estimation of the expectation values $\langle . \rangle_{clamped}$ and $\langle . \rangle_{free}$ requires enumerating all possible state configurations. For efficiency, we use Gibbs sampling [96], a Markov Chain Monte Carlo method, to approximate their values.

4.4.5 BM Inference for MLIR Ranking

For a new query q and the retrieved documents D_q, the relevance probability of a document $d_i \in D_q$ can be estimated by $P(sd_i, q) = \sum_{\mathbf{sd}_q \setminus sd_i, \mathbf{st}_q} P(\mathbf{sd}_q, \mathbf{st}_q, q)$. It is then straightforward to determine $\hat{l}_i = \operatorname{argmax}_{sd_i} P(sd_i, q)$ as the rank label for ranking and use the value of $P(\hat{l}_i, q)$ to break the tie. However, exact estimation of $P(sd_i, q)$ is time-consuming as enumeration of all possible global states is required. For efficient online prediction, we use *mean field* approximation [37] for the inference. Mean field theory has solid foundation based on variational principle. Here we simply present the procedure of the mean field approximation for BM, and leave the formal justifications to [37].

In mean field approximation, the state distribution of each node only relies on the states of its neighbors which are all fixed to their *average state value*. Thus, giving the machine, we have the following equations:

$$P(sd_i = r) = \frac{\exp\left[\sum_j \vec{\theta} \cdot \vec{\psi}(d_i, t_j)\langle st_j \rangle r + \vec{w} \cdot \vec{\phi}(q, d_i) r\right]}{\sum_r \exp\left[\sum_j \vec{\theta} \cdot \vec{\psi}(d_i, t_j)\langle st_j \rangle r + \vec{w} \cdot \vec{\phi}(q, d_i) r\right]} \tag{4.13}$$

$$P(st_j = r) = \frac{\exp\left[\sum_i \vec{\theta} \cdot \vec{\psi}(d_i, t_j) r \langle sd_i \rangle\right]}{\sum_r \exp\left[\sum_i \vec{\theta} \cdot \vec{\psi}(d_i, t_j) r \langle sd_i \rangle\right]} \tag{4.14}$$

$$\langle sd_i \rangle = \sum_r P(sd_i = r)r \tag{4.15}$$

$$\langle st_j \rangle = \sum_r P(st_j = r)r \tag{4.16}$$

where $P(sd_i = r)$ is the probability of document d_i with the rank label (i.e., node state) r, $P(st_j = r)$ is the probability of topic t_j with the rank label r, $\langle sd_i \rangle$ is the average rank label (i.e., average node state) of document d_i, and $\langle st_j \rangle$ is the average rank label of topic t_j. Equation 4.13 computes the relevance probability of a document given the average rank labels of all the topics. Similarly, Equation 4.14 computes the relevance probability of a topic given the average rank labels of all the documents. Equation. 4.15 and 4.16 estimate the average rank labels given the probability distributions computed by Equation 4.13 and 4.14.

Equations 4.13–4.16 are called mean field equations, and can be solved using the following iterative procedure for a fixed-point solution:

1. Assume an average state value for every node;
2. For each node, estimate its state value probability using Equation 4.13 and 4.14 given the average state values (i.e., average rank labels) of its neighbors;
3. Update the average state values for each node using Equation 4.15 and 4.16;
4. Go to step 2 until the average state values converge.

Each iteration requires $O(|T_q| + |D_q|)$ time, and is linear to the number of nodes.

4.4.6 BM Training with MAP Optimization

In the previous sections, BM is optimized for rank label prediction. However, rank label prediction is just loosely related to MLIR accuracy as the exact relevance labels are not necessary for deriving the correct ranking orders. [101] presents a ranking model which directly optimizes IR evaluation measure; and the best ranking performance has been reported. Therefore, we will train our model in a similar way, i.e., optimizing the Mean Average Precision (MAP) of MLIR.

We know that the predicted ranking order is produced by $\pi(q, D_q, F)$. The average precision for q is defined as follows:

$$AvgP_q = \frac{\sum_{i=1}^{n(q)} p_q(i) y_i}{\sum_{i=1}^{n(q)} y_i}$$

where $n(q)$ is the number of retrieved documents, y_i is assigned as 1 or 0 depending on whether $d_{i'}$ is relevant or not ($d_{i'}$ is the document ranked at the i-th position, i.e., $\pi(d_{i'}) = i$), and $p_q(i)$ is the precision at the rank position of i: $p_q(i) = \frac{1}{i}\sum_{j<i} y_j$. Suppose N is the number of queries, MAP is the mean of average precision over all the queries, i.e., $MAP = \frac{1}{N}\sum_q AvgP_q$.

Simply optimizing MAP is likely to cause overfitting. Instead, we try to maximize the following revised objective function:

$$MAP - C\sum_{x=1}^{X} ||w_x||^2 - C\sum_{y=1}^{Y} ||\theta_y||^2 \tag{4.17}$$

where w_x and θ_y are the weights of node features and edge features respectively, so the last two terms are *L-2* regularization terms representing the complexity of the model. The function is therefore a tradeoff between the model's accuracy and complexity controlled by coefficient C. SVM-MAP [101] used a similar function to minimize a linear combination of the same *L-2* norm with the hinge loss relaxation of MAP loss.

Since MAP is not a continuous function with the weights of the BM, Powell's Direction Set Method [73], which does not involve derivation computations, is used for the optimization. To achieve optimal performance, Powell's method is repeatedly called a number of times with different initial values of the BM's weights each time. One particular set of the initial values takes the weight values learned when the

BM is trained to optimize classification accuracy in Section 4.4.4. The mean field approximation (Section 4.4.5) is used in model inference as well.

4.5 Experiments and Results

We evaluated the proposed MLIR ranking algorithms. The experiments were conducted on two datasets: (1) TREC-5&6 English-Chinese CLIR data; (2) Chinese and English multilingual Web search data. The baseline was the ranking score combination algorithm, referred to as *ScoreComb* below. In this method, different ranking algorithms including Ranking SVM and SVM-MAP were first used to learn ranking functions for Chinese and English documents separately. The scores were then combined by a log linear model following [84, 87].

Three prevalent learning algorithms, i.e., SVC (SVM classifier with probability estimation), RSVM (Ranking SVM), and SVM-MAP, were used to compare the performance of MLIR ranking. These algorithms represent three typical categories of ranking schemes: (1) SVC is a typical classification-based ranking algorithm; (2) RSVM is the state-of-the-art ranking algorithm based on pair-wise preference order classification; (3) SVM-MAP is a ranking algorithm directly optimizing IR relevance measure. We used the source codes of LibSVM[1], SVM-Light[2] and SVM-map[3] to run SVC, RSVM and SVM-MAP, respectively.

The proposed BM classifier (BMC) and BM classifier with MAP optimizer (BMC-MAP) were evaluated and compared against the above algorithms. In order to directly assess the contribution of the relevance among documents, we reduced BMC and BMC-MAP into the conventional log linear models by simply removing the hidden

---

[1] `http://www.csie.ntu.edu.tw/~jlin/libsvm`

[2] `http://svmlight.joachims.org/`

[3] `http://projects.yisongyue.com/svmmap/`

units and the edges. This produced two additional systems, namely LOG and LOG-MAP, for our comparative study.

4.5.1 Experiments on TREC CLIR Data

In this section, we study the contribution of similarities between documents and topics on CLIR. The CLIR task of TREC-5&6 was defined as using English queries to retrieve Chinese documents. Although multilingual result merge was not required, it was valuable to study the effectiveness of relevant English documents in improving cross-lingual relevance estimation for Chinese documents. Since the joint ranking model required English retrieval, we additionally indexed the English TIPSTER corpus from LDC[4]. We use query CH1-28 in TREC-5 topics for training and CH29-54 in TREC-6 topics for testing.

Three free machine translation engines were used to translate English queries to Chinese, and then an BM25 (Okapi) model [79] was employed for Chinese document retrieval based on the combined query translations. For learning the ranking models, we implemented 25 commonly used query-document relevance features described in the literature [57] using the translated queries, including the scores of TFIDF, BM25, and language modeling IR, etc.

To create BM for joint relevance ranking, 500 English documents were retrieved from TIPSTER using the original query. We ranked them by their BM25 scores. Since there was no relevance annotation in the English documents, we chose 20 documents and assigned them with one of the following two labels: 0 for the last 10 documents in the result; and 1 for the top 10. That is, we assumed the first 10 documents as relevant and the last 10 as irrelevant according to BM25 in the source language. During both training and inference, the states of the English document nodes were

---

[4]LDC Catalog No.: LDC93T3A. `http://www.ldc.upenn.edu/Catalog/`

Table 4.1. TREC-6 CLIR performance by 11-point precision-recall and AP measure

recall	BM25	SVC	RSVM	SVM-MAP	LOG	BMC	LOG-MAP	BMC-MAP
0	0.658	0.736	0.788	0.798	0.715	0.796	0.797	0.815
0.1	0.495	0.476	0.531	0.598	0.475	0.583	0.592	0.591
0.2	0.411	0.393	0.427	0.486	0.391	0.469	0.480	0.502
0.3	0.345	0.354	0.385	0.414	0.349	0.412	0.411	0.423
0.4	0.289	0.324	0.346	0.368	0.324	0.367	0.366	0.376
0.5	0.251	0.282	0.299	0.316	0.281	0.312	0.315	0.323
0.6	0.203	0.222	0.241	0.245	0.214	0.247	0.241	0.269
0.7	0.164	0.174	0.200	0.185	0.175	0.183	0.182	0.220
0.8	0.074	0.099	0.101	0.086	0.099	0.088	0.084	0.107
0.9	0.010	0.020	0.027	0.016	0.018	0.017	0.016	0.030
1.0	0.002	0.007	0.012	0.006	0.004	0.007	0.006	0.008
AP	0.249	0.253	0.280	0.301	0.250	0.299	0.299	0.314

fixed to one of the above values. The 25 relevance features were also calculated for English documents based on the source queries.

The CLIR results are given in Table 4.1 in terms of average precision (AP) and 11-point precision-recall measures. Since no multilingual result merge was involved, the BM25 score between the translated queries and the Chinese documents was used as the baseline of ranking. It is obvious that all the learning algorithms outperformed the baseline. Furthermore, SVM-MAP outperformed RSVM and SVC, and BMC-MAP outperformed BMC, implying that like monolingual ranking direct optimization of IR measure is also critical to CLIR ranking.

We further conducted t-test, which showed that BMC significantly outperformed LOG ($p = 0.009$) and RSVM ($p = 0.011$). This indicates that utilizing monolingual IR results for CLIR ranking is effective. The AP improvement from SVM-MAP and LOG-MAP to BMC-MAP was not as large as from LOG to BMC. This may be caused by optimizing Equation 4.17 using a Boltzmann machine. Different from SVM-MAP training which achieved global optimum, BMC-MAP training only achieved a sub-optimal solution. However, although suffered from under-training, BMC-MAP still significantly outperformed SVM-MAP by 4.15% ($p = 0.032$).

4.5.2 MLIR Experiments on Web Search Data

In this section, we discuss the experiments and results on real Web search dataset from a commercial search engine. We will first describe the Web search dataset we used, and then give the details of experiments of multilingual runs.

4.5.2.1 Multilingual Web Search Data.

Our Web search data consisted of queries and returned Web pages from query logs of a commercial search engine. There were two separate monolingual query logs for English and Chinese. The retrieved Web pages were annotated with ratings from 0 (irrelevant) to 5 (perfect) by human labelers, representing different relevance levels (or rank labels). About 600 features were associated with English pages, and about 900 features were associated with Chinese pages. The features consisted of query-dependent features (e.g. BM25) and query-independent features (e.g. PageRank)[5]. The English part of the corpora was previously used by several learning-to-rank studies in monolingual IR [10, 12].

For multilingual ranking, we manually selected 1,000 queries from the English query log and their translations from the Chinese query log. Thus we obtained 1,000 pairs of bilingual queries. Based on these queries and their labeled results, we constructed a bilingual ranking corpus: Given an input query, the corresponding Chinese and English Web pages associated with the rank labels were put together. The resulting corpus contained 17,791 Chinese and 32,049 English pages with manual rank labels. After applying the rules to unify the two sets of features (see Section 4.3.3), we ended up with 352 features in the unified bilingual feature space.

In addition, the edge features specific to our joint model, i.e., the 12 similarities measuring the correlations between documents and salient topics, were also computed.

---

[5]To protect copyright of the data which belong to a commercial search engine, we cannot disclose the detailed definition of these features.

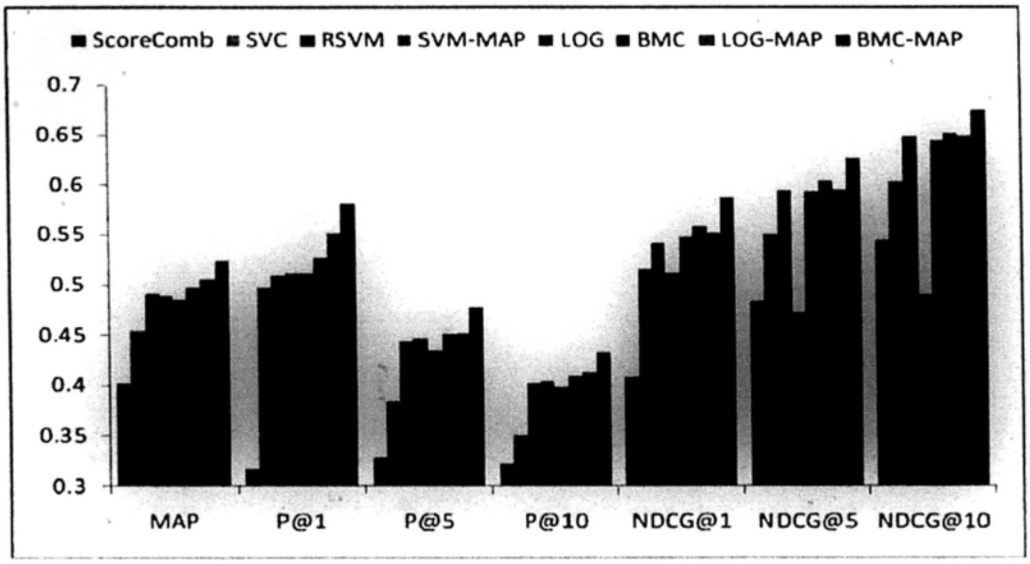

Figure 4.4. Comparison of ranking results using multilingual Web search data.

All the model parameters were tuned on a development set with 197 queries; and 803 queries were used for 4-fold cross validation.

4.5.2.2 Experiments on Multilingual Ranking.

The results of MAP, precision@1,5,10 and NDCG@1,5,10 are presented in Figure 4.4. Apparently, all the models learned using the multilingual feature space outperformed the *ScoreComb* baseline. The t-test showed that all improvements were statistically significant ($p < 0.05$). This confirmed the advantage of the learning-to-rank approaches over the score combination approach in directly learning a ranking function from the given features.

By optimizing the ranking order of document pairs, RSVM was expected to perform better than SVC. This was verfied by our MLIR experiments. Similar to the TREC result, BMC achieved comparable results with RSVM, implying that classification-based ranking algorithms, by making use of the relevance among individual documents (i.e., the document-topic similarities), performed equally well with the state-of-the-art ranking models. Interestingly, SVM-MAP underperformed

Table 4.2. The comparison results of using and without using clusters in BM models.

	MAP	P@1	P@5	P@10	NDCG@1	NDCG@5	NDCG@10
LOG	0.484	0.511	0.435	0.397	0.546	0.591	0.641
BMC	0.497	0.527	0.451	0.409	0.557	0.604	0.651
LOG-MAP	0.504	0.552	0.452	0.413	0.551	0.594	0.649
BMC-MAP	0.523	0.580	0.478	0.432	0.587	0.626	0.674

RSVM. This may be because SVM-MAP cannot support the fine-grained 6-level relevance while RSVM can.

BMC-MAP outperformed all other models. In terms of MAP, it outperformed the baseline by 30.22% ($p = 0.003$), SVC by 15.12% ($p = 0.006$), BMC by 5.33% ($p = 0.029$), RSVM by 3.90% ($p = 0.023$), and SVM-MAP by 7.40% ($p = 0.009$). Table 4.2 showed the enhancement from the joint ranking model by comparing BMC with LOG, as well as BMC-MAP with LOG-MAP. The p-value on MAP difference was 0.04 between BMC and LOG, and is 0.027 between BMC-MAP and LOG-MAP, implying the significant contribution of the correlations among documents.

Further analysis on the clustering effects showed that our method could group documents with similar relevance labels from different languages into the same clusters. The documents with different ratings were either put into different clusters or not included in the salient clusters. To assess the effects of clustering, we calculated the averaged relevance difference over every pair of documents in each cluster based on human relevance judgement. The relevance difference of two documents was defined as $\frac{|l_1 - l_2|}{l_{max} - l_{min}}$ where l_1 and l_2 were the rank labels, and $l_{max}(l_{min})$ the max (min) rating level defined. The averaged difference in a salient cluster reflected the purity of the cluster according to human judgment. It was observed that around 70% of the clusters had the values of averaged relevance difference less than 0.3 (i.e., for 0-5 rating levels, it meant the difference among document ratings was less than 1 in average), which indicated that clustering was very effective in identifying documents with similar relevance.

4.6 Chapter Summary

We studied how to rank Web documents of different languages based on their relevance to the query. Different from existing researches which focused on relevance score combination, we applied the learning-to-rank approach to MLIR ranking. To further improve multilingual ranking accuracy, a joint ranking model was proposed. The model exploited various similarities among documents in addition to the commonly used query-document relevance features. By using Boltzmann machine, this new model first uncovered salient topics among retrieved documents, and then collaboratively identified relevant documents and relevant topics.

The joint ranking model makes use of search results retrieved for common search interests, i.e., bilingual queries. This justifies our hypothesis that in addition to query formulation like CLQS, the correlation or commonality derived from common search interests are also helpful to multilingual and cross-lingual search ranking problems.

CHAPTER 5

MONOLINGUAL RANKING WITH CROSS-LINGUAL INFORMATION

In the previous chapters, we have presented new techniques based on the observation of common search interests of users in different languages for cross-language and multilingual Web search applications. But machine translation (MT) is required to translate search results into user's language for browsing. Contemporary MT technologies cannot cater for user's smooth reading behavior effectively. Due to this predicament, the proposed cross-language and multilingual search technologies cannot unleash the power of their full potential. Therefore, before a perfect MT system appears (if this would ever happen), we propose to re-investigate the potential of monolingual search to capture and cater for user's cross-lingual information needs.

Web search quality can vary widely across languages, even for the same information need. In this chapter, we propose to exploit this variation in quality by learning a ranking function on bilingual queries, i.e., queries representing equivalent search interests in the query logs of different languages. For a given bilingual query, along with its corresponding monolingual query logs and monolingual ranking, we generate a ranking on pairs of documents, one from each language based on click-through data. We then learn a linear ranking function which exploits bilingual features (similarities) on the document pairs, as well as standard monolingual features of individual documents. Finally, we show how to reconstruct monolingual ranking from the learned bilingual ranking. Using publicly available Chinese and English query logs, we demonstrate that for both languages our technique leads to significant improvements over a state-of-the-art monolingual ranking algorithm.

The materials presented in this chapter are partially based on our work presented in [27].

5.1 Introduction

Web search quality can vary widely across languages due to the origin of the search terms. For example, ranking search results for the query "托马斯 霍布斯" (Thomas Hobbes) is more difficult in Chinese than it is in English, even while holding the basic ranking function constant. By the same token, ranking search results for the query "Han Feizi" (韩非子) is likely to be harder in English than in Chinese. A large portion of Web queries have such properties that they are originated in a language different from the one they are searched.

This variance in problem difficulty across languages is not unique to Web search; it appears in a wide range of natural language processing (NLP) problems. Much recent work on using bilingual data has focused on exploiting these variations in difficulty across languages to improve a variety of monolingual tasks, including parsing [36, 88, 11, 91], named entity recognition [14], and topic clustering [99]. In this work, we exploit a similar intuition to improve *monolingual* Web search.

Our problem setting differs from cross-lingual Web search, where the goal is to return machine-translated results from one language in response to a query from another [52]. We operate under the assumption that for many monolingual English queries (e.g., "Han Feizi"), there exist good documents on English Web, which are nevertheless ranked behind many irrelevant pages due to different reasons, such as their unpopularity or the sparseness of click-through data due to lack of exposure. If we have Chinese information as well, we can exploit it to help find these English documents. As we will see, machine translation can provide important predictive information in our monolingual setting, but we do not have to rely on machine translation for displaying the output to the user.

We approach our problem by learning a ranking function for *bilingual queries* – queries that are easily translated (e.g., with machine translation) and appear in the query logs of two languages (e.g., English and Chinese). Due to this property, bilingual queries reflect user's cross-lingual information needs even though a monolingual system would not actually search in a different language. Given query logs in both languages, we identify bilingual queries with sufficient click-through statistics in both sides. Large-scale aggregated click-through data were proved useful and effective in learning ranking functions [19]. Using these statistics on clicked documents, we can construct a ranking corpus over *pairs* of documents, one from each language. Given a bilingual query, we use this ranking to learn a linear scoring function on pairs of documents.

We find that our bilingual rankings have good monolingual ranking properties. In particular, given an optimal pairwise bilingual ranking, we show that simple heuristics can effectively approximate the optimal monolingual ranking. Using these heuristics and our learned pairwise scoring function, we can derive a ranking for new, unseen bilingual queries. We develop and test our bilingual ranker on English and Chinese with two large, publicly available query logs from the AOL search engine (English query log) [69] and the Sougou search engine (Chinese query log) [58]. For both languages, we achieve significant improvements over monolingual Ranking SVM (RSVM) baselines [34, 41], which exploit a variety of monolingual features.

This chapter is organized as follows: Section 5.2 presents our ranking model learned from bilingual information derived from the query logs; Section 5.3 describes the feature and similarity measures used for learning; Section 5.4 discusses the experiments and results; finally, we summarize the chapter in Section 5.5.

5.2 Learning to Rank Using Bilingual Information

Given a set of bilingual queries, we now describe how to learn a ranking function for monolingual data that exploits information from both languages. Our procedure has three steps: Given two monolingual rankings, we construct a *bilingual* ranking on pairs of documents, one from each language. Then we learn a linear scoring function for pairs of documents that exploits monolingual information (in both languages) and bilingual information. Finally, given this ranking function on pairs and a new bilingual query, we reconstruct a monolingual ranking for the language of interest. This section addresses these steps in turn.

5.2.1 Bilingual Training Data

Given a set of bilingual queries and their retrieved documents, we try to learn a ranking function by using the bilingual information among the documents as constraints for better ranking search results of document in the desired language. Without loss of generality, suppose we rank English documents with constraints from Chinese documents. Hence we simply call the Chinese part constraint documents. Given an English log L_e and a Chinese log L_c, our ranking algorithm takes as input a bilingual query pair $q = (q_e, q_c)$ where $q_e \in L_e$ and $q_c \in L_c$, a set of returned English documents $\{e_i\}_{i=1}^{N}$ from q_e, and a set of constraint Chinese documents $\{c_j\}_{j=1}^{n}$ from q_c. In order to create bilingual ranking data, we first generate monolingual ranking data from click-through statistics. For each language-query-document triple, we calculate the aggregated click count across all users and rank documents according to this statistic. We denote the count of a page as $C(e_i)$ or $C(c_j)$.

The use of click-through statistics as feedback for learning ranking functions is not without controversy, but recent empirical results on large data sets suggest that the aggregated user clicks provides an informative indicator of relevance preference for a query. Joachims et al. [42] showed that *relative* feedback signals generated

Table 5.1. Click-through data of a bilingual query pair extracted from query logs.

Bilingual query pair (***Mazda***, 马自达)		
doc	URL	click #
e_1	`www.mazda.com`	229
e_2	`www.mazdausa.com`	185
e_3	`www.mazda.co.uk`	5
e_4	`www.starmazda.com`	2
e_5	`www.mazdamotosports.com`	2
	......	
c_1	`www.faw-mazda.com`	50
c_2	`price.pcauto.com.cn/brand.jsp?bid=17`	43
c_3	`auto.sina.com.cn/salon/FORD/MAZDA.shtml`	20
c_4	`car.autohome.com.cn/brand/119/`	18
c_5	`jsp.auto.sohu.com/view/brand-bid-263.html`	9
	......	

from clicks correspond well with human judgments. Dou et al. [19] revealed that a straightforward use of *aggregated clicks* could achieve better ranking than using explicitly labeled data because click-through data contained fine-grained differences between documents, which were useful for learning an accurate and reliable ranking function. Therefore, we leverage aggregated clicks for comparing the relevance order of documents. Note that there is nothing specific to our technique that requires click-through statistics. Indeed, our methods could easily be employed with human annotated data. Table 5.1 gives an example of a bilingual query pair and the aggregated click count of each result page.

Given two monolingual documents, a preference order can be inferred if one document is clicked more often than the other. To allow for cross-lingual information, we extend the order of individual documents into that of *bilingual document pairs*: given two bilingual document pairs, we will write $\left(e_i^{(1)}, c_j^{(1)}\right) \succ \left(e_i^{(2)}, c_j^{(2)}\right)$ to indicate that the pair of $\left(e_i^{(1)}, c_j^{(1)}\right)$ is ranked higher than the pair of $\left(e_i^{(2)}, c_j^{(2)}\right)$.

Definition 1 $\left(e_i^{(1)}, c_j^{(1)}\right) \succ \left(e_i^{(2)}, c_j^{(2)}\right)$ *if and only if one of the following relations hold:*

1. $C(e_i^{(1)}) > C(e_i^{(2)})$ *and* $C(c_j^{(1)}) \geq C(c_j^{(2)})$

2. $C(e_i^{(1)}) \geq C(e_i^{(2)})$ *and* $C(c_j^{(1)}) > C(c_j^{(2)})$

Note, however, that from a purely monolingual perspective, this definition introduces orderings on documents that may not have initially existed. For English ranking, for example, we may have $\left(e_i^{(1)}, c_j^{(1)}\right) \succ \left(e_i^{(2)}, c_j^{(2)}\right)$ even when $C(e_i^{(1)}) = C(e_i^{(2)})$. This leads us to the following asymmetric definition of $\succ$ that we use in practice:

Definition 2 $\left(e_i^{(1)}, c_j^{(1)}\right) \succ \left(e_i^{(2)}, c_j^{(2)}\right)$ *if and only if the following relation holds:* $C(e_i^{(1)}) > C(e_i^{(2)})$ *and* $C(c_j^{(1)}) \geq C(c_j^{(2)})$

With this definition, we can unambiguously compare the relevance of bilingual document pairs based on the order of monolingual documents. The advantages are two-fold: (1) we can treat multiple cross-lingual document similarities the same way as the commonly used query-document features in a uniform manner of learning; (2) with the similarities, the relevance estimation on bilingual document pairs can be enhanced, and this in return can improve document ranking.

5.2.2 Ranking Model

Given a pair of bilingual queries (q_e, q_c), we can extract the set of corresponding bilingual document pairs and their click counts $\{(e_i, c_j), (C(e_i), C(c_j))\}$, where $i = 1, \ldots, N$ and $j = 1, \ldots, n$. Based on that, we produce a set of bilingual ranking instances $S = \{\mathbf{\Phi_{ij}}, z_{ij}\}$, where each $\mathbf{\Phi_{ij}} = \{\mathbf{x_i}; \mathbf{y_j}; \mathbf{s_{ij}}\}$ is the feature vector of (e_i, c_j). consisting of three components: $\mathbf{x_i} = \mathbf{f}(q_e, e_i)$ is the vector of monolingual relevance features of e_i, $\mathbf{y_i} = \mathbf{f}(q_c, c_j)$ is the vector of monolingual relevance features of c_j, and $\mathbf{s_{ij}} = \mathbf{sim}(e_i, c_j)$ is the vector of cross-lingual similarities between e_i and c_j, and $z_{ij} = (C(e_i), C(c_j))$ is the corresponding click counts.

The task is to select the optimal function that minimizes a given loss with respect to the order of ranked bilingual document pairs and the gold order. We resort to Ranking SVM (RSVM) [34, 41] learning for classification on pairs of instances. Compared with the baseline RSVM (monolingual), our algorithm learns to classify on *pairs of bilingual document pairs* rather than on pairs of individual documents.

Let f being a linear function:

$$f_{\vec{w}}(e_i, c_j) = \vec{w}_x \cdot \mathbf{x_i} + \vec{w}_y \cdot \mathbf{y_j} + \vec{w}_s \cdot \mathbf{s_{ij}} \tag{5.1}$$

where $\vec{w} = \{\vec{w}_x; \vec{w}_y; \vec{w}_s\}$ denotes the weight vector, in which the elements correspond to the relevance features and similarities. For any two bilingual document pairs, their preference relation is measured by the difference of the functional values of Equation 5.1:

$$\begin{aligned}
\left(e_i^{(1)}, c_j^{(1)}\right) \succ \left(e_i^{(2)}, c_j^{(2)}\right) & \quad \Leftrightarrow \\
f_{\vec{w}}\left(e_i^{(1)}, c_j^{(1)}\right) - f_{\vec{w}}\left(e_i^{(2)}, c_j^{(2)}\right) > 0 & \quad \Leftrightarrow \\
\vec{w}_x \cdot \left(\mathbf{x_i}^{(1)} - \mathbf{x_i}^{(2)}\right) + \vec{w}_y \cdot \left(\mathbf{y_j}^{(1)} - \mathbf{y_j}^{(2)}\right) + \vec{w}_s \cdot \left(\mathbf{s_{ij}}^{(1)} - \mathbf{s_{ij}}^{(2)}\right) > 0 &
\end{aligned}$$

We then create a new training corpus based on the preference ordering of any two such pairs: $S' = \{\mathbf{\Phi}'_{\mathbf{ij}}, z'_{ij}\}$, where the new feature vector becomes

$$\mathbf{\Phi}'_{\mathbf{ij}} = \left\{\mathbf{x_i}^{(1)} - \mathbf{x_i}^{(2)}; \mathbf{y_j}^{(1)} - \mathbf{y_j}^{(2)}; \mathbf{s_{ij}}^{(1)} - \mathbf{s_{ij}}^{(2)}\right\}, \tag{5.2}$$

and the class label

$$z'_{ij} = \begin{cases} +1, & \text{if} \quad \left(e_i^{(1)}, c_j^{(1)}\right) \succ \left(e_i^{(2)}, c_j^{(2)}\right); \\ -1, & \text{if} \quad \left(e_i^{(2)}, c_j^{(2)}\right) \succ \left(e_i^{(1)}, c_j^{(1)}\right) \end{cases}$$

is a binary preference value depending on the order of bilingual document pairs. The problem is to solve SVM objective: $\min_{\vec{w}} \frac{1}{2}\|\vec{w}\|^2 + \lambda \sum_i \sum_j \xi_{ij}$, subject to bilingual constraints: $z'_{ij} \cdot (\vec{w} \cdot \mathbf{\Phi}'_{\mathbf{ij}}) \geq 1 - \xi_{ij}$ and $\xi_{ij} \geq 0$, where $1 \leq i \leq N$, $1 \leq j \leq n$, and

ξ_{ij} is a slack variable measuring the degree of pairwise misclassification of preference order.

There are potentially $\Gamma = nN$ bilingual document pairs for each query, and the number of comparable pairs may be much larger due to the combinatorial nature (but less than $\Gamma(\Gamma - 1)/2$). To speed up training, we resort to stochastic gradient descent (SGD) optimizer [86] to approximate the true gradient of the loss function evaluated on a single instance (i.e., per constraint). The parameters are then adjusted by an amount proportional to this approximate gradient. For large data set, SGD-RSVM can be much faster than batch-mode gradient descent.

5.2.3 Inference

The solution $\vec{w}$ forms a vector orthogonal to the hyper-plane of RSVM. To predict the order of bilingual document pairs, the ranking score can simply be calculated by Equation 5.1. However, a prominent problem is how to derive the full order of monolingual documents for output from the order of bilingual document pairs. To our knowledge, there is no precise conversion algorithm in polynomial time. We thus adopt two heuristics for approximating the real document score:

- **H-1 (max score):** Choose the maximum score of the pair as the score of document, i.e., $score(e_i) = max_j(f(e_i, c_j))$.
- **H-2 (mean score):** Average over all the scores of pairs associated with the ranked document as the score of this document, i.e., $score(e_i) = 1/n \sum_j f(e_i, c_j)$.

Intuitively, for the rank score of a single document, **H-2** combines the "voting" scores from its n constraint documents that are weighted equally, and **H-1** simply chooses the maximum one. A formal yet time-consuming approach to the problem is to make use of rank aggregation formalism [20, 59], which will be left to our future work. The two simple heuristics are employed here because of their simplicity and

Table 5.2. List of monolingual relevance measures used as IR features in our model.

IR Feature	Description
BM25	Okapi BM25 score [76]
BM25_PRF	Okapi BM25 score with pseudo-relevance feedback [77]
LM_DIR	Language-model-based IR score with Dirichlet smoothing [103]
LM_JM	Language-model-based IR score with Jelinek-Mercer smoothing [103]
LM_ABS	Language-model-based IR score with absolute discounting [103]
PageRank	PageRank score [6]

efficiency. The time complexity of the approximation is linear to the number of documents to rank.

5.3 Features and Similarities

Standard features for learning to rank include differet query-document features, e.g., BM25 [76], as well as query-independent features, e.g., PageRank [6]. Our feature space consists of both these standard monolingual features and cross-lingual similarities among documents. Cross-lingual similarities are measured by either different translation mechanisms, e.g., dictionary-based translation or machine translation, or even without using any translation at all.

5.3.1 Monolingual Relevancy Features

In learning to rank, the relevancy between query and documents and the measures based on link analysis are commonly used as features [57]. The discussion on their details is beyond the scope of this work. We implement six most typical features as shown in Table 5.2. These include sets of measures such as BM25, language-model-based IR score, and PageRank. Since most conventional IR and Web search relevance measures fall into this category, collectively they are referred to as *IR features* hereafter. Note that for a given bilingual document pair (e, c), the monolingual IR features consist of relevance score vectors $\mathbf{f}(q_e, e)$ in English and $\mathbf{f}(q_c, c)$ in Chinese.

5.3.2 Cross-lingual Document Similarities

To measure the document similarity across different languages, we define the similarity vector $\mathbf{sim}(e, c)$ as a series of functions mapping a bilingual document pair to positive real numbers. Intuitively, a good similarity function is one which maps a set of cross-lingual relevant documents into close scores and maintains a large distance between irrelevant documents. Four categories of similarity measures are employed.

Dictionary-based Similarity (DIC): For dictionary-based document translation, we use the similarity measure proposed by Mathieu et al. [63]. Given a bilingual dictionary, we let $T(e, c)$ denote the set of word pairs (w_e, w_c) such that w_e is a word in English document e, w_c is a word in Chinese document c, and w_e is the English translation of w_c. We define $tf(w_e, e)$ and $tf(w_c, c)$ to be the term frequency of w_e in e and that of w_c in c respectively. Let $df(w_e)$ and $df(w_c)$ be the English document frequency for w_e and Chinese document frequency for w_c respectively. If n_e (n_c) is the total number of English (Chinese), then the bilingual idf is defined as $idf(w_e, w_c) = \log \frac{n_e+n_c}{df(w_e)+df(w_c)}$. The cross-lingual document similarity $sim(e, c)$ is calculated by Equation 4.11 (see Sect. 4.4.3).

Similarity Based on Machine Translation (MT): For machine translation, cross-lingual measure is equivalent to the monolingual similarity between one document and the translation of the other. We therefore adopt cosine function for this directly [82].

Translation Ratio (RATIO): Translation ratio is defined as two sets of ratios of translatable terms using a bilingual dictionary: RATIO_FOR – the percentage of words in e which can be translated to words in c; RATIO_BACK – the percentage of words in c which can be translated back to words in e.

URL LCS Ratio (URL): The ratio of Longest Common Subsequence [17] between the URLs of two pages being compared. This measure is useful to capture pages in different languages with similar URLs, such as `www.airbus.com`, `www.airbus.com.cn`, etc.

Note that each set of similarities above except URL includes 3 values based on different fields of Web page: title, body, and title+body.

5.4 Experiments and Results

5.4.1 Evaluation Metric

Ranking metrics, such as mean average precision [8] and Normalized Discounted Cumulative Gain [38], designed for data sets with human relevance judgment are widely used for evaluation. However, human labeled data are not readily available to us. Therefore, we use the Kendall's tau coefficient [43, 41] to measure the degree of correlation between two rankings. For simplicity, we assume strict orderings of any given ranking and ignore all pairs with ties (instances with the identical click count). Kendall's tau is defined as $\tau(r_a, r_b) = (P - Q)/(P + Q)$, where P is the number of concordant pairs and Q is the number of disconcordant pairs in the given orderings r_a and r_b. The value is a real number within $[-1, +1]$, where -1 indicates a complete inversion, and $+1$ stands for perfect agreement, and zero indicates no correlation.

Existing ranking techniques heavily depend on human relevance judgment that is very costly to obtain. Similar to Dou et al. [19], our method utilizes automatically aggregated click count in query logs as the gold standard ranking for deriving the order of relevancy for evaluation, but not like their work, we use the click-through of different languages. We average Kendall's tau values between the algorithm output and the gold standard based on click frequency for all test queries.

5.4.2 Data Sets

Query logs can be the basis for constructing high quality ranking corpus. Due to the proprietary issue of log, no public ranking corpus based on real-world search engine

Table 5.3. Statistics on AOL and Sogou query logs.

	AOL(EN)	Sogou(CH)
# sessions	657,426	5,131,000
# unique queries	10,154,743	3,117,902
# clicked queries	4,811,650	3,117,590
# clicked URLs	1,632,788	8,627,174
time span	2006/03-05	2006/08
size	2.12GB	1.56GB

log is currently available. Moreover, to build a predictable bilingual ranking corpus, the logs of different languages are required and have to meet certain conditions: (1) they should be large so that a good number of bilingual query pairs could be identified; (2) for the identified query pairs, there should be statistically reasonable amount of associated click-through information; and (3) the click frequency should be well distributed at both sides so that the preference order between bilingual document pairs can be derived for SVM learning.

For these reasons, we used two independent and publicly accessible query logs to construct our bilingual ranking corpus: English AOL log[1] and Chinese Sogou log[2]. Table 5.3 shows some statistics of these two large query logs.

We automatically identified 10,544 bilingual query pairs from the two logs using the Java API for Google Translate[3], in which each query had certain number of clicked URLs. To better control bilingual equivalency of queries, we ensured the bilingual queries in each of these pairs are bi-directional translations. We then downloaded all their clicked pages, which resulted in 70,180 English[4] and 111,197 Chinese documents.

---

[1] `http://gregsadetsky.com/aol-data/`

[2] `http://www.sogou.com/labs/dl/q.html`

[3] `http://code.google.com/p/google-api-translate-java/`

[4] AOL log only recorded the domain portion of the clicked URLs, which misled document downloading. We used the "search within site or domain" function of a major search engine to approximate the real clicked URLs by keeping the first returned result for each query.

These documents formed two independent collections, which were indexed separately for retrieval and feature calculation.

For good quality, it was necessary to have sufficient click-through data for each query. Thus we further identified 1,084 out of 10,544 bilingual query pairs, in which each query had at least 10 clicked and downloadable documents. This smaller collection was used to cross-validate our model, containing 21,711 English and 28,578 Chinese documents[5]. In order to compute cross-lingual document similarities based on machine translation (see Section 5.3.2), we automatically translate all these 50,298 documents using Google Translate, i.e., English to Chinese and vice versa. Then the bilingual document pairs are constructed, and all the monolingual features and cross-lingual similarities are computed (see Section 5.3.1 and 5.3.2).

5.4.3 English Ranking Performance

Here we examined the ranking performance of our English ranker under different similarity settings. We used traditional RSVM [34, 41] without any bilingual consideration as the *baseline*, which used English IR features only. We conducted this experiment using all the 1,084 bilingual query pairs with 4-fold cross validation (each fold with 271 query pairs). The number of constraint documents n was empirically set to 5. The results are shown in Table 5.4.

Bilingual constraints were helpful to improve English ranking. Our pairwise settings unanimously outperformed the RSVM baseline. The paired two-tailed t-test [90] showed that most improvements resulted from heuristic **H-2** (mean score) were statistically significant at 99% confidence level ($p<0.01$). Relatively fewer significant improvements were made by heuristic **H-1** (max score). This was because the maximum pair score was just a rough approximation to the optimal document score. But

[5]Since Sogou log had more clicked URLs, for balancing with the number of English pages, we kept at most 50 pages per Chinese query.

Table 5.4. Kendall's tau values of English ranking. The significant improvements over the baseline (99% confidence) are represented in boldface with the p-values given in parenthesis. * indicates significant improvement over IR (no similarity). Note that $n = 5$.

Models	Pair	H-1 (max)	H-2 (mean)
RSVM (baseline)	n/a	0.2424	0.2424
IR (no similarity)	0.2783	0.2445	0.2445
IR+DIC	0.2909	0.2453	0.2496
IR+MT	0.2858	**0.2488*** (p=0.0003)	**0.2494*** (p=0.0004)
IR+DIC+MT	0.2901	0.2481	**0.2514*** (p=0.0009)
IR+DIC+RATIO	0.2946	0.2466	**0.2519*** (p=0.0004)
IR+DIC+MT +RATIO	0.2940	**0.2473*** (p=0.0009)	**0.2539*** (p=1.5e-5)
IR+DIC+MT +RATIO+URL	0.2979	**0.2533*** (p=2.2e-5)	**0.2577*** (p=4.4e-7)

this simple scheme worked surprisingly well and still consistently outperformed the baseline.

Note that our bilingual model with only IR features, i.e., IR (no similarity), also outperformed the baseline. This was because this setting involved IR features of n constraint Chinese documents in addition to the IR features of English documents in the baseline.

The DIC similarity did not work as effectively as MT. This may be due to the problems such as out-of-vocabulary words and translation ambiguity, which were common in static bilingual dictionary. These issues however could be better dealt with by MT. When DIC was combined with RATIO, which included both forward and backward translation of words, it could capture the correlation between bilingually similar pages. For this reason, its performance was improved.

We find that URL similarity, although simple, was very effective and outperformed the case without URL similarity by 1.5–2.4% of Kendall's tau value than. This was because the URLs of the top Chinese (constraint) documents were often similar to

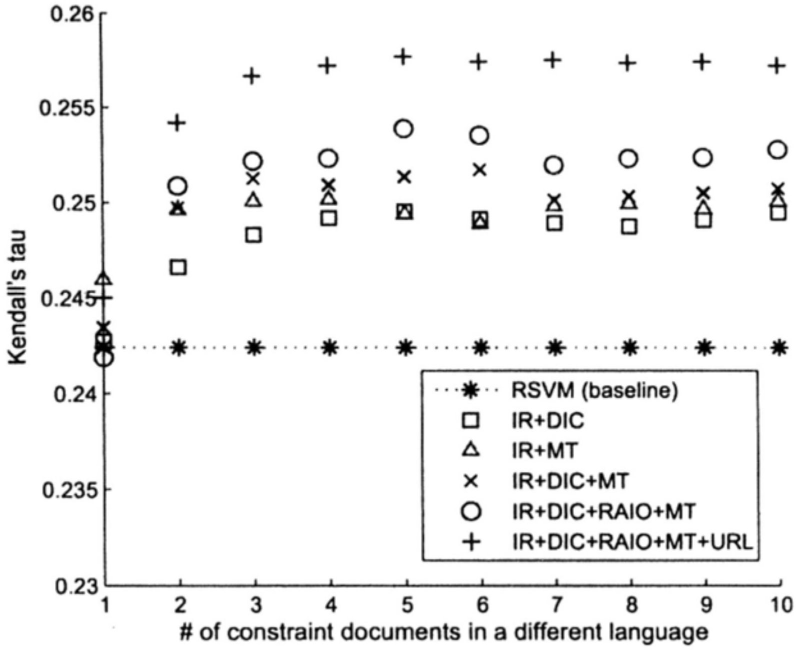

Figure 5.1. English ranking results vary with the number of constraint Chinese documents.

many of the returned English URLs which were generally more regular. For example, in query pair (Toyota Camry, 丰田佳美), 9 out of 13 English pages were anchored by the URLs containing keywords "toyota" and/or "camry", and 3 out of 5 constraint documents' URLs also contained them. In contrast, the URLs of returned Chinese pages were less regular in general. This also explained why this measure did not improve much for Chinese ranking (see Section 5.4.4).

We also varied the parameter n to study how the performance changed with different number of constraint Chinese documents. Figure 5.1 showed the results using heuristic **H-2**. More constraint documents were generally helpful. But when only one constraint document was used, some feature would be detrimental to the ranking under some feature settings. One explanation was that the document clicked most often may not be relevant, and thus it is likely that similar English pages would not exist for the first Chinese page. Joachims et al. [42] found that users' click behavior was biased by the rank of search engine at the first and/or second positions

Table 5.5. Kendall's tau values of Chinese ranking. The significant improvements over the baseline (99% confidence) are represented in boldface with the p-values given in parenthesis. * indicates significant improvement over IR (no similarity). Note that $n = 5$.

Models	Pair	H-1 (max)	H-2 (mean)
RSVM (baseline)	n/a	0.2935	0.2935
IR (no similarity)	0.3201	0.2938	0.2938
IR+DIC	0.3220	**0.2970** (p=0.0060)	**0.2973*** (p=0.0020)
IR+MT	0.3299	**0.2992*** (p=0.0034)	**0.3008*** (p=0.0003)
IR+DIC+MT	0.3295	**0.2991*** (p=0.0014)	**0.3004*** (p=0.0008)
IR+DIC+RATIO	0.3240	**0.2972*** (p=0.0010)	**0.2968*** (p=0.0014)
IR+DIC+MT +RATIO	0.3303	**0.2973*** (p=0.0004)	**0.3007*** (p=0.0002)
IR+DIC+MT +RATIO+URL	0.3288	**0.2981*** (p=0.0005)	**0.3024*** (p=1.5e-6)

(especially the first). More constraint pages would be helpful as the pages after the first were less biased and the click counts could reflect relevancy more accurately.

5.4.4 Chinese Ranking Performance

We also evaluated Chinese ranking with English constraint documents under similar configurations as Section 5.4.3. The results are shown in Table 5.5 and Figure 5.2.

Table 5.5 showed that improvements in Chinese ranking were encouraging. Kendall's tau values under all the settings were significantly better than not only the baseline but also IR (no similarity). This suggested that English information was generally more helpful to Chinese ranking than the other way around. This was because there were a high proportion of Chinese queries having English or foreign-language origins in our dataset. For these queries, relevant information at the Chinese side was relatively poorer, so English ranking was more reliable. As far as we can, we manually identified 215 such queries from all the 1,084 bilingual queries (amount to 23.2%).

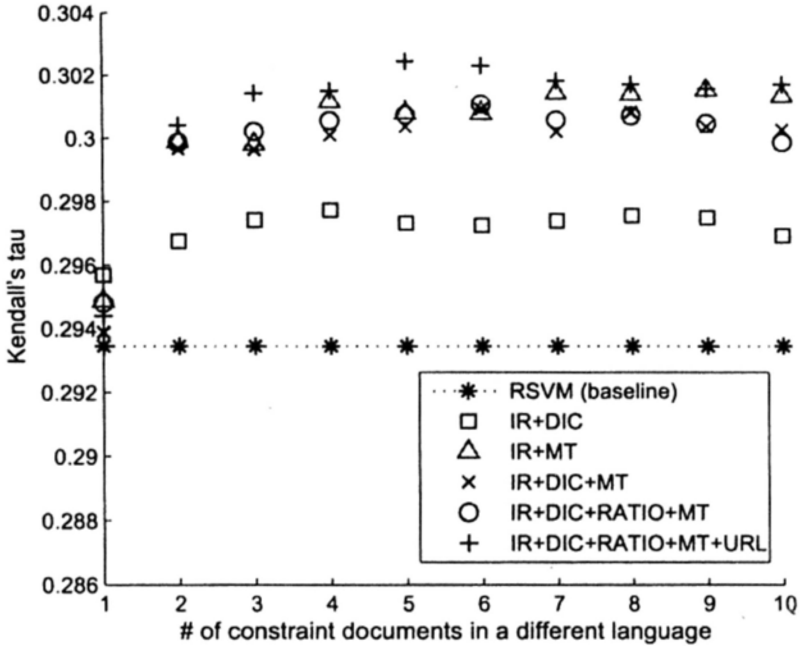

Figure 5.2. Chinese ranking results vary with the number of constraint English documents.

To shed more light on this finding, we examined top-20 queries improved most by our method (with all features and similarities) over the baseline. As shown in Table 5.6, most of the top improved Chinese queries were about concepts originated from English or other languages, or something non-local (bolded). Interestingly, among these Chinese queries, "政治漫画" (political catoons) was improved most by English ranking. Note that this topic was considered rare (or sensitive) content on the Chinese Web. In contrast, we found this type of queries very few in most improved English queries. But we could still observe "Bruce Lee" (李小龙), a Chinese Kung Fu actor, and "peony" (牡丹), the national flower of China, whose information was more popular on the Chinese Web and thus helpful to English ranking. As for the exceptions like "Sunrider" (仙妮蕾德) and "Aniston" (安妮斯顿), despite their English origins, we observed that they had surprisingly sparse click counts in the English log. But they appeared to be more interesting to Chinese users who provided a lot more click-through data that were found in the Chinese log.

Table 5.6. Top 20 most improved bilingual queries. Bolded words mean positive example based on our hypothesis. * marks an exception.

Most improved Chinese queries	Most improved English queries
沙门氏菌 (salmonella)	free online tv (免费在线电视)
苏格兰 (scotland)	weapons (武器)
咖啡因 (caffeine)	lily (百合)
墓志铭 (epitaph)	cable (电缆)
英国历史 (british history)	*sunrider (仙妮蕾德)
政治漫画 (political cartoons)	*aniston (安妮斯顿)
免疫系统 (immune system)	clothes (衣服)
葡萄酒瓶 (wine bottles)	*three little pigs (三只小猪)
匈牙利 (hungary)	hair care (护发)
巫术 (witchcraft)	neon (霓虹灯)
爆米花 (popcorn)	**bruce lee (李小龙)**
脓疱疮 (impetigo)	radish (萝卜)
卫生间设计 (bathroom design)	chile (智利)
鸽子 (pigeon)	**peony (牡丹)**
北极熊 (polar bear)	toothache (牙痛)
非洲地图 (map of africa)	free online translation (免费在线翻译)
拉布拉多犬 (labrador retriever)	water (水)
帕米拉安德森 (pamela anderson)	oil (石油)
瑜伽服装 (yoga clothing)	shopping network (购物网)
联邦快递 (federal express)	*prince harry (哈里王子)

5.5 Chapter Summary

In this chapter, we aim to improve monolingual Web search ranking for bilingual queries, by exploiting bilingual information derived from click-through logs of different languages. The thrust of our technique is to use search ranking of one language and cross-lingual information to help ranking of another language. Our pairwise ranking scheme based on bilingual document pairs can easily integrate different kinds of similarities into the existing framework and significantly improves both English and Chinese ranking performance.

CHAPTER 6
CONCLUSIONS

6.1 Conclusions

Query formulation and relevance ranking are the core search engine components. Search for information across different languages is especially challenging due to the predicaments in overcoming language barriers. The special characteristic of Web environment provides rich resources and knowledge, which can be useful for effective cross-language search. In this dissertation, we have explored the use of common search interests across different languages in formulating queries and ranking documents for better serving user's cross-lingual information needs.

Defining exactly what makes a search interest common across the languages is difficult because no precise cross-lingual similarity measure is available for queries from different languages. Because of this, we defined a subset of common search interests, referred to as *bilingual queries*, which were automatically derived from two monolingual query logs of real-world search engines using machine translation (Chapter 2). We found that a large proportion of frequently issued queries were bilingual. Thus, it was inferred in general that a significant part of search interests are common across different languages. To justify the usefulness of this observed property, this research has made the following contributions to Web search:

1. In the level of query formulation (Chapter 3), we developed effective algorithms for learning to suggest closely related queries across different languages, referred to as cross-lingual query suggestion (CLQS). Our method differs from existing approaches for query suggestion and for query translation in three aspects:

- We extended monolingual query suggestion to cross-lingual query suggestion. To our knowledge, this is the first attempt in this direction.
- We leveraged the target-language query log to suggest more cohesive complete queries than by using a query translation approach.
- We proposed a discriminative method to learn to estimate cross-lingual query similarity instead of manually define such a measure. This enabled us not only to obtain a more suitable similarity measure, but also to adapt the approach to different language pairs more easily.

In our experiments, we have compared our approach with several baseline methods. The baseline CLQS system applied a typical query translation approach, using a bilingual dictionary with co-occurrence-based translation disambiguation. Benchmarked under French-English and Chinese-English settings, this baseline approach only covered 10-15% of the relevant queries suggested by an monolingual query suggestion system (when the exact translation of the original query was given). By leveraging additional resources such as parallel corpora, Web mining and query log-based monolingual query suggestion, the final system covered 42-44% of the relevant queries suggested by a monolingual query suggestion system with precision as high as 79.6% and 93.8% for French-English and Chinese-English tests, respectively.

To further evaluate the quality of the suggested queries, CLQS system was used as a query "translation" system in the CLIR tasks. Using the TREC-6 French-English and NTCIR-4 Chinese-English CLIR tasks as benchmarks, CLQS consistently demonstrated higher effectiveness than traditional query translation methods using either bilingual dictionary or state-of-the-art statistical machine translation approaches. Three traditional information retrieval models, i.e., BM25, language modeling, and TFIDF vector space model, were adopted in the experiments.

The improvement on TREC-6 French-English CLIR task by using CLQS demonstrated the high quality of the suggested queries. This also implied the strong correspondence between the input French queries and the English queries in the log. For queries of Chinese and English which showed weaker correspondence in the log, CLQS performed surprisingly well due to the comprehensive bilingual data resources and the satisfactory coverage of the query logs.

Pseudo-relevance feedback (PRF) and CLQS both expanded the original query for improving CLIR performance. But they exploited different types of resources and distinctive mechanisms, and therefore could be complementary to each other. Interestingly, for French-English CLIR, the complementary effect from the pseudo feedback to CLQS was relatively smaller than that for Chinese-English CLIR. This was because French-English CLQS could suggest closely related queries more effectively from the English query log than the Chinese-English case due to the stronger correspondence between the search interests of users in French and English.

2. To generalize the usefulness of common search interests, we then studied how to rank Web documents of different languages based on their relevance to the query (Chapter 4). This work was done by using the correlation information among retrieval results derived from common search interests.

   Different from existing researches which focused on relevance score combination, we adopted the learning-to-rank approach to MLIR ranking first. By constructing a unified multilingual feature space, popular ranking algorithms, such as Ranking SVM and SVM-MAP, were applied to MLIR ranking, and they significantly outperformed the score combination approach.

To further improve MLIR ranking accuracy, a joint ranking model was proposed. The model exploited various similarities among documents in addition to the commonly used query-document relevance features. By using multilingual clustering, this new method first uncovered salient topics among retrieved documents; and then learned to collaboratively identify relevant documents and relevant topics using a probabilistic graphic model called Boltzmann machine. By modeling the similarity among search results, Boltzmann machine leveraged relevant documents in one language to help the relevance estimation of documents in different languages, and induced the joint relevance probability for all the documents. Benchmarks using TREC-5&6 CLIR datasets and a multilingual ranking dataset from a search engine showed that effectiveness of the corresponding CLIR and MLIR ranking tasks were significantly improved by the joint ranking model.

3. We proposed to improve monolingual Web search by using bilingual click-through information derived from common search interests found in the query logs (Chapter 5). This technique aimed to enable users to meet their important portion of cross-lingual information needs without having to depend on machine-translated search results. For a given bilingual query, with the corresponding monolingual query log and monolingual ranking, we generated a ranking corpus based on pairs of documents, one from each language. We then learned a ranking function that incorporated bilingual features of document pairs as well as monolingual features of individual documents. Finally, we reconstructed monolingual ranking from the learned bilingual ranking.

   The thrust of this technique is based on similar intuition in joint ranking (Chapter 4), i.e., to use search ranking of one language and cross-lingual information to help ranking of another language. But it is important to note that this approach did not rely on any human relevance judgement which is costly to obtain. Instead, this pairwise ranking scheme was based on bilingual document pairs whose preference orders were derived automatically from click-through data. Moreover, this approach can easily integrate different kinds of cross-lingual similarities into the existing framework. Using publicly available Chinese and English query logs, we demonstrated that for both languages our ranking technique exploiting bilingual data led to significant improvements over the state-of-the-art monolingual ranking algorithm.

6.2 Future Work

As far as we are concerned, this is the first research effort made in this new direction on discovering common search interests shared by users of different languages and applying this knowledge to information retrieval applications on the Web. Due to the common knowledge potentially existing across many languages, the techniques that exploit this information are expected to have broad applications for Web search. In the future, we can further our study in the following directions:

1. We have exploited several types of monolingual and cross-lingual information in cross-lingual query suggestion. However, more types of information can be integrated into the general framework for the estimation of cross-lingual query similarity. This is an interesting improvement for our future work. Improvements can also be made in the way to determine similar queries. For example, query popularity or click counts of queries can be explicitly taken into consideration so that the most popular (thus usual) query formulations can be suggested. Furthermore, we can take into account user factors, such as search sessions of particular users, to better identify their search intents for disambiguating query's meaning. For instance, in order to identify which sense the word "apple" is in a given query, the feedback information in previous sessions

of the same user can be utilized, which contain document clicks as to the pages of fruit or company.

2. One of the key advantages of query logs is that they are up-to-date in terms of user needs and vocabulary. But our method works well on standard text collections that are not necessarily aligned with the timeframe of query logs. This may be because our query log is newer, in which all queries were issued in the year of 2005, than the collections of news, which happened in 1988-90 and 1998-99. Our log is characterized with good backward compatibility with the news previously occurred as we found that nearly all the topics about the test queries can correlate to some entries in the English query log. On the other hand, our log also contains the queries that turned out to become very popular later on. For example, although far from so popular as nowadays, queries on "Barack Obama" still frequently appear in this query log of early days. This suggests that query logs may have large intemporal value as a lexical resource. We would like to study specifically the temporal issues of exploiting query logs for query suggestion in future work.

3. The joint ranking model is in fact a generic ranking mechanism. It is not specifically applicable to MLIR. Thus its contribution to monolingual IR should be studied in the future. Besides content similarity, any types of relationship among Web pages, such as structural similarity, hyperlink relation, etc., could be used to improve ranking under this framework. Moreover, Boltzmann machine training to optimize IR evaluation measure, i.e., mean average precision, only achieved a sub-optimal solution. Therefore, there are rooms for further improvement. Finally, the inference speed is one of the major concerns in adopting the joint ranking models for real Web search. This renders future research work on methods to speed up the ranking process, such as offline document clustering.

4. Another interesting direction is to continue the study of using bilingual information to improve the effectiveness of monolingual Web search. We will study the recovery of the optimal document ordering from pairwise ordering using well-founded formalism such as rank aggregation approaches [20, 59]. Furthermore, to reduce our reliance on bilingual resources for producing cross-lingual features, we may involve more sophisticated monolingual features that do not transfer cross-lingually but are asymmetric for either side, such as clustering or document classification features, which can be built from human-edited domain taxonomies like the open directory project DMOZ[1].

5. At last, but not the least, we believe ranking adaptation from one language to another is a promising direction for our future work. Oftentimes, it is costly or prohibitive to prepare training data for search engine of each language, especially when fast deployment of a ranking model is required. Thus, it is important to reuse the valuable monolingual training data under different language domains. We will investigate adapting the ranking model trained in one language to the ranking task in different languages based on the similar intuition as using common knowledge across languages.

[1] `http://www.dmoz.org/`

BIBLIOGRAPHY

[1] D. H. Ackley, G. E. Hinton, and T. J. A. Sejnowski. Learning algorithm for boltzmann machines. *Cognitive Science*, 9:147–169, 1985.

[2] V. Ambati and U. Rohini. Using monolingual clickthrough data to build cross-lingual search systems. In *Proceedings of ACM SIGIR Workshop on New Directions in Multilingual Information Access*, 2006.

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

[4] L. A. Ballesteros and W. B. 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, 1998.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems*, pages 107–117, 1998.

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

[8] C. Buckley and E. M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 33–40, 2000.

[9] R. Bunescu and R. J. Mooney. Collective information extraction with relational Markov networks. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 438–445, 2004.

[10] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 89–96, 2005.

[11] D. Burkett and D. Klein. Two languages are better than one (for syntactic parsing). In *Proceedings of the 2008 Conference on Empirical Methods on Natural Language Processing*, pages 877–886, 2008.

[12] Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon. Adapting ranking SVM to document retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 186–193, 2006.

[13] C. C. Chang and C. Lin. LibSVM: a library for support vector machines (version 2.3). 2001. `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`

[14] M.-W. Chang, D. Goldwasser, D. Roth, and Y. Tu. Unsupervised constraint driven learning for transliteration discovery. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 299–307, 2009.

[15] H.-H. Chen, M.-S. Lin, and Y.-C. Wei. Novel association measures using web search with double checking. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 1009–1016, 2006.

[16] P.-J. Cheng, J.-W. Teng, R.-C. Chen, J.-H. Wang, W.-H. Lu, and L.-F. Chien. Translating unknown queries with web corpora for cross-language information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 146–153, 2004.

[17] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms (2nd Edition)*. McGraw-Hill Inc., 2001.

[18] H. Cui, J. R. Wen, J.-Y. Nie, and W.-Y. Ma. Query expansion by mining user logs. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):829–839, 2003.

[19] Z. Dou, R. Song, X. Yuan, and J.-R. Wen. Are click-through data adequate for learning web search rankings? In *Proceeding of the 17th ACM conference on Information and knowledge management*, pages 73–82, 2008.

[20] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th international conference on World Wide Web*, pages 613–622, 2001.

[21] E. A. Fox and J. A. Shaw. Combination of multiple searches. In *Proceedings of the 2nd Text REtrieval Conference (TREC-2), National Institute of Standards and Technology*, 1994.

[22] Y. Freund, R. Iyer, R. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2004.

[23] A. Fuji and T. Ishikawa. Applying machine translation to two-stage cross-language information retrieval. In *Proceedings of 4th Conference of the Association for Machine Translation in the Americas (AMTA)*, pages 13–24, 2000.

[24] J. F. Gao, M. Li, A. Wu, and C.-N. Huang. Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics*, 31(4):531–574, 2005.

[25] J. F. Gao, J.-Y. Nie, H. He, W. Chen, and M. Zhou. Resolving query translation ambiguity using a decaying co-occurrence model and syntactic dependence relations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 183–190, 2002.

[26] J. F. Gao, J.-Y. Nie, E. Xun, J. Zhang, M. Zhou, and C. Huang. Improving query translation for CLIR using statistical models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 96–104, 2001.

[27] W. Gao, J. Blitzer, M. Zhou, and K.-F. Wong. Exploiting bilingual information to improve web search. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*, pages 1075–1083, 2009.

[28] W. Gao, C. Niu, J.-Y. Nie, M. Zhou, J. Hu, K.-F. Wong, and H.-W. Hon. Cross-lingual query suggestion using query logs of different languages. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 463–470, 2007.

[29] W. Gao, C. Niu, J.-Y. Nie, M. Zhou, K.-F. Wong, and H.-W. Hon. Exploiting query logs for cross-lingual query suggestion. *ACM Transactions on Information Systems*, 2010 (accepted for publication).

[30] W. Gao, C. Niu, M. Zhou, and K.-F. Wong. Joint ranking for multilingual web search. In *Proceedings of the 31st European Conference on Information Retrieval Research*, pages 114–125, 2009.

[31] D. Gleich and L. Zhukov. Svd subspace projections for term suggestion ranking and clustering. In *Technical Report, Yahoo! Research Labs*, 2004.

[32] G. Grefenstette and J. Nioche. Estimation of English and non-English language use on the www. In *Proceedings of Recherche d'Information Assistée par Ordinateur (RIAO)*, pages 237–246, 2000.

[33] R. Herbrich, T. Graepel, and K. Obermayer. *Large Margin Rank Boundaries for Ordinal Regression.* MIT Press, Cambridge, 2000.

[34] R. Herbrich, T. Graepel, and K. Obermayer. *Large Margin Rank Boundaries for Ordinal Regression.* The MIT Press, 2000.

[35] D. 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, 1993.

[36] R. Hwa, P. Resnik, A. Weinberg, C. Cabezas, and O.Kolak. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11(3):311–325, 2005.

[37] T. S. Jaakkola. *Variational Methods for Inference and Estimation in Graphical Models*. PhD thesis, MIT, 1997.

[38] K. Jarvelin and J. Kekalainen. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 41–48, 2000.

[39] J. Jeon, W. B. Croft, and J. Lee. Finding similar questions in large question and answer archives. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pages 84–90, 2005.

[40] M.-G. Jiang, S. H. Myaeng, and S. Y. Park. Using mutual information to resolve query translation ambiguities and query term weighting. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*; pages 223–229, 1999.

[41] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142, 2002.

[42] T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Transaction on Information Systems*, 25(2), 2007.

[43] M. Kendall. A new measure of rank correlation. *Biometrika*, 30:81–89, 1938.

[44] K. Kishida, K.-H. Chen, S. Lee, K. Kuriyama, N. Kando, H.-H. Chen, S. H. Myaeng, and K. Eguchi. Overview of CLIR task at the 4th NTCIR workshop. In *Proceedings of 4th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pages 1–59, 2004.

[45] J. Ko, S. Luo, and E. Nyberg. A probabilistic graphical model for joint answer ranking in question answering. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 343–350, 2007.

[46] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit*, pages 79–86, 2005.

[47] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (demo)*, pages 177–180, 2007.

[48] P. Koehn, F. J. Och, and D. Marcu. Statistical phrase based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54, 2003.

[49] W. Kraaij, J.-Y. Nie, and M. Simard. Embedding web-based statistical translation models in cross-language information retrieval. *Computational Linguistics*, 29(3):381–419, 2003.

[50] S. Kullback. *Information Theory and Statistics*. John Wiley & Sons Press, NY, 1959.

[51] K. L. Kwok, S. Choi, and N. Dinstl. Rich results from poor resources: NTCIR-4 monolingual and cross-lingual retrieval of korean texts using Chinese and English. *ACM Transactions on Asian Language Information Processing*, 4(2):136–162, 2005.

[52] V. Lavrenko, M. Choquette, and W. B. Croft. Cross-lingual relevance models. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 175–182, 2002.

[53] J. H. Lee. Combining the evidence of different relevance feedback methods for information retrieval. *Information Processing and Management*, 34(6):681–691, 2001.

[54] C.-H. Li, D. Zhang, M. Li, M. Zhou, M. Li, and Y. Guan. A probabilistic approach to syntax-based reordering for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 720–727, 2007.

[55] W.-C. Lin and H.-H. Chen. Merging mechanisms in multilingual information retrieval. In *Advances in Cross-Language Information Retrieval, the 3rd Workshop of the Cross-Language Evaluation Forum, CLEF 2002*, volume 2785 of *Lecture Notes in Computer Science, Springer*, 2003.

[56] T.-Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, 2009.

[57] T.-Y. Liu, J. Xu, T. Qin, W. Xiong, and H. Li. Letor: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of ACM Workshop on Learning to Rank for Information Retrieval*, 2007.

[58] Y. Liu, Y. Fu, M. Zhang, S. Ma, and L. Ru. Automatic search engine performance evaluation with click-through data analysis. In *Proceedings of the 16th International Conference on World Wide Web*, pages 1133–1134, 2007.

[59] Y.-T. Liu, T.-Y. Liu, T. Qin, Z.-M. Ma, and H. Li. Supervised rank aggregation. In *Proceedings of the 16th International Conference on World Wide Web*, pages 481–489, 2007.

[60] F. López-Ostenero, J. Gonzalo, and F. Verdejo. Noun phrases as building blocks for cross-language search assistance. *Information Processing and Management*, 41:549–568, 2005.

[61] W.-H. Lu, L.-F. Chien, and H.-J. Lee. Anchor text mining for translation extraction of query terms. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 388–389, 2001.

[62] B. Mathieu, R. Besancon, and C. Fluhr. Multilingual document clusters discovery. In *Proceedings of Recherche d'Information Assistée par Ordinateur (RIAO)*, pages 1–10, 2004.

[63] B. Mathieu, R. Besancon, and C. Fluhr. Multilingual document clusters discovery. In *Proceedings of Recherche d'Information Assistée par Ordinateur (RIAO)*, pages 1–10, 2004.

[64] 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, 2002.

[65] C. Monz and B. J. Dorr. Iterative translation disambiguation for cross-language information retrieval. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 520–527, 2005.

[66] J.-Y. Nie, M. Simard, P. Isabelle, and R. 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, 1999.

[67] F. J. Och. *Statistical Machine Translation: From Single-Word Models to Alignment Templates*. PhD thesis, RWTH Aachen, Germany, 2002.

[68] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[69] G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems (INFOSCALE)*, 2006.

[70] C. Peters and P. Sheridan. Multilingual information access. In F. C. M. Agosti and G. Pasi, editors, *Lecture Notes in Information Retrieval*, pages 51–80. Springer, 2000.

[71] A. Pirkola, T. Hedlund, H. Keshusalo, and K. Jarvelin. Dictionary-based cross-language information retrieval: Problems, methods, and research findings. *Information Retrieval*, 4(3/4):209–230, 2001.

[72] J. Ponte and W. B. 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, 1998.

[73] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing (§10.5).* Cambridge University Press, 1992.

[74] S. E. Roberson. The probability ranking principle in IR. *Journal of Documentation*, 33(4):294–304, 1977.

[75] S. E. Robertson. On term selection for query expansion. *Journal of Documentation*, 46:359–364, 1990.

[76] S. E. Robertson. Overview of the okapi projects. *Journal of Documentation*, 53(1):3–7, 1997.

[77] S. E. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of the American Society of Information Science*, 27(3):129–146, 1976.

[78] S. E. Robertson and S. Walker. Some simple effective approximation to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 232–241, 1994.

[79] S. E. Robertson, S. Walker, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference (TREC-3)*, pages 200–225, 1995.

[80] J. J. Rocchio. Relevance feedback information retrieval. In G. Salton, editor, *The Smart Retrieval System – Experiments in Automatic Document Processing*, pages 313–323. Prentice-Hall, Englewood Cliffs, N.J., 1971.

[81] R. Salakhutdinov, A. Mnih, and G. Hinton. Restricted boltzmann machines for collaborative filtering. In *Proceedings of 24th International Conference on Machine Learning*, pages 791–798, 2007.

[82] G. Salton. *Automatic Text Processing.* Addison-Wesley Publishing, 1998.

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

[84] J. Savoy and P. Y. Berger. Selection and merging strategies for multilingual information retrieval. In *Multilingual Information Access for Text, Speech and Images, 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004*, volume 3491 of *Lecture Notes in Computer Science, Springer*, pages 27–37. Springer-Verlag, 2005.

[85] P. Schauble and P. Sheridan. Cross-language information retrieval (CLIR) track overview. In *Proceedings of the 6th Text REtrieval Conference (TREC-6)*, pages 31–44, 2000.

[86] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for SVM. In *Proceedings of the 24th International Conference on Machine Learning*, pages 807–814, 2007.

[87] L. Si and J. A. Callan. Clef 2005: Multilingual retrieval by combining multiple multilingual ranked lists. In *Accessing Multilingual Information Repositories, 6th Workshop of the Cross-Language Evalution Forum, CLEF 2005*, volume 4022 of *Lecture Notes in Computer Science, Springer*, pages 121–130. Springer-Verlag, 2006.

[88] D. A. Smith and N. A. Smith. Bilingual parsing with factored estimation: Using English to parse Korean. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, 2004.

[89] A. J. Smola and B. A. Scholkopf. Tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004.

[90] M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pages 623–632, 2007.

[91] B. Snyder and R. Barzilay. Unsupervised multilingual learning for morphological segmentation. In *Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics*, pages 737–745, 2008.

[92] M. Taylor, J. Guiver, S. Robertson, and T. Minka. Softrank: Optimising non-smooth rank metrics. In *Proceedings of the International Conference on Web Search and Web Data Mining*, pages 77–86, 2008.

[93] M.-F. Tsai, T.-Y. Liu, T. Qin, H.-H. Chen, and W.-Y. Ma. Frank: A ranking method with fidelity loss. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 383–390, 2007.

[94] M.-F. Tsai, Y.-T. Wang, and H.-H. Chen. A study of learning a merge model for multilingual information retrieval. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 195–202, 2008.

[95] E. Voorhees, N. K. Gupta, and B. Johnson-Laird. The collection fusion problem. In *Proceedings of the 3rd Text REtrieval Conference (TREC-3)*, pages 95–104, 1994.

[96] B. Walsh. Markov chain monte carlo and gibbs sampling. In *Lecture Notes for EEB 596z*, 2002. `http://nitro.biosci.arizona.edu/courses/EEB596/handouts/Gibbs.pdf`

[97] J. R. Wen, J.-Y. Nie, and H. J. Zhang. Query clustering using user logs. *ACM Transactions on Information Systems*, 20(1):59–81, 2002.

[98] R. W. White, C. L. A. Clarke, and S. Cucerzan. Comparing query logs and pseudo-relevance feedback for web-search query refinement. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 831–832, 2007.

[99] Y. Wu and D. W. Oard. Bilingual topic aspect classification with a few training examples. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 203–210, 2008.

[100] J. Xu and H. Li. Adarank: A boosting algorithm for information retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 391–398, 2007.

[101] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 271–278, 2007.

[102] C. X. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the 10th International Conference on Information and Knowledge Management*, pages 403–410, 2001.

[103] C. X. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 334–342, 2001.

[104] D. Zhang, M. Li, N. Duan, C.-H. Li, and M. Zhou. Measure word generation for English-Chinese SMT systems. In *Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics*, pages 89–96, 2008.

[105] Y. Zhang and P. 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, 2004.