Technical Report

UCAM-CL-TR-616
ISSN 1476-2986

Number 616

UNIVERSITY OF CAMBRIDGE

Computer Laboratory

Dictionary characteristics in cross-language information retrieval

Donnla Nic Gearailt

February 2005

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500

http://www.cl.cam.ac.uk/

This technical report is based on a dissertation submitted February 2003 by the author for the degree of Doctor of Philosophy to the University of Cambridge, Gonville and Caius College.

Technical reports published by the University of Cambridge Computer Laboratory are freely available via the Internet:

http://www.cl.cam.ac.uk/TechReports/

ISSN 1476-2986

Abstract

In the absence of resources such a as suitable MT system, translation in Cross-Language Information Retrieval (CLIR) consists primarily of mapping query terms to a semantically equivalent representation in the target language. This can be accomplished by looking up each term in a simple bilingual dictionary. The main problem here is deciding which of the translations provided by the dictionary for each query term should be included in the query translation. We tackled this problem by examining different characteristics of the system dictionary. We found that dictionary properties such as scale (the average number of translations per term), translation repetition (providing the same translation for a term more than once in a dictionary entry, for example, for different senses of a term), and dictionary coverage rate (the percentage of query terms for which the dictionary provides a translation) can have a profound effect on retrieval performance. Dictionary properties were explored in a series of carefully controlled tests, designed to evaluate specific hypotheses. These experiments showed that (a) contrary to expectation, smaller scale dictionaries resulted in better performance than large-scale ones, and (b) when appropriately managed e.g. through strategies to ensure adequate translational coverage, dictionary-based CLIR could perform as well as other CLIR methods discussed in the literature. Our experiments showed that it is possible to implement an effective CLIR system with no resources other than the system dictionary itself, provided this dictionary is chosen with careful examination of its characteristics, removing any dependency on outside resources.

Acknowledgments

Many thanks are due to my supervisor, Karen Spärck Jones, whose help and guidance throughout the last number of years has been greatly appreciated. Thanks are also due to Ted Briscoe for his input on various aspects of the project. Thank to Aline, Anna, Sylvia, Naila and Advaith for lots of fun times in the lab, and a huge big thanks to my husband Miles who provided much-need reassurance and advice. This work was supported by European Commission Training and Mobility of Researchers Category 20 Grant No. ERBFMBICT972453. Many thanks to them for their sponsorship.

Contents

List of Figures

Chapter 1

Introduction

A few years ago, I decided I wanted to find the telephone number of a friend who was living in France. I knew that the French National Telephone Directory was available on-line. So I opened up Netscape Navigator and went to the Altavista search engine [1]. I typed in the words *France Telecom Directory*, expecting to find the relevant web site somewhere near the top of the retrieved document list.

Strangely, it wasn't there. I refined my search a little more. I used the advanced features. I added some optional keywords. I repeated the process several more times. Eventually, grinding my teeth with frustration, I gave up.

Several hours later, I was telling a friend about this over a cup of coffee. She said, "Oh, no! You should have typed *annuaire electronique*".

1.1 The Need for Effective Multilingual Information Retrieval

As the anecdote above illustrates, standard information retrieval does not deal particularly well with multilingual document collections.

In the past, this was not really a problem. However, behind the huge amount of hype surrounding the supposed "dawn of the information age" is a real need for multilingual electronic document management. It is clear that new techniques are required. Multilingual information retrieval has become an active area of research in the last decade. Larger investigations include those sponsored by the US TREC conferences [40] and CLEF [74]. We are interested in the subfield of multilingual IR that we call *Cross-Language Information Retrieval* (CLIR) - where a user query in one language is matched against a document collection in another.

This chapter is organised as follows. First, we present a very basic model of information retrieval (IR). This is included for readers unfamiliar with the field of IR to acquaint them with IR concepts mentioned in later sections - real-world systems are much more complex than the model described below. Then, we define Cross-Language IR (CLIR) and discuss how CLIR differs from the monolingual task, and explain further why research in this area is warranted. Following this, we outline the importance of dictionary-based CLIR within the CLIR field. Finally, we explain our approach of investigating dictionary-based CLIR in a resource-poor environment, classifying our investigations into four categories and giving a summary of our results for each category. The chapter explains why the work described here is significant and concludes with an outline of the contents of the remainder of this thesis.

1.2 Basics of Information Retrieval

As stated above, this section is included primarily for the benefit of readers hitherto unfamiliar with the field of IR. We present some very basic retrieval concepts that need to be defined in order to discuss Cross Language IR further on. Modern day IR systems are much more advanced than this basic model - a good introduction to modern IR systems and their complexities may be found in Spärck Jones and Willett 1997 [96].

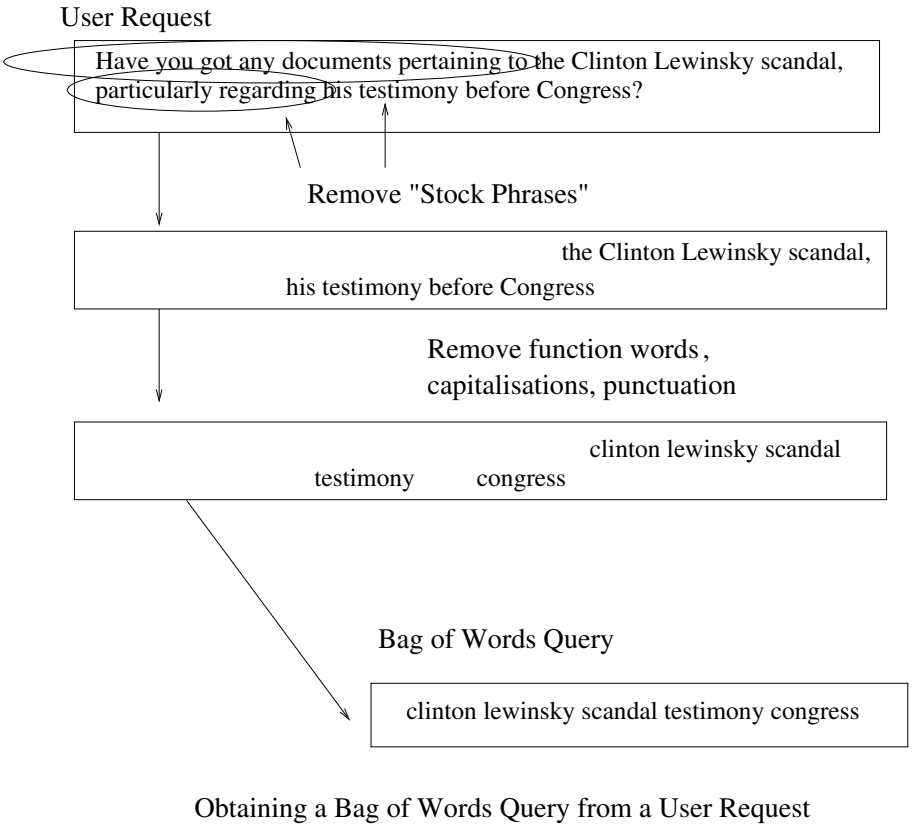

Figure 1.1: Processing a Request to Form a Bag of Words Query

An *Information Retrieval System* is defined as any system that matches a *user request* against a *document collection*, returning a list of *documents* considered relevant to the request. The user request is an expression of a user information need. Traditionally, users made such requests to a professional librarian, who would then suggest reading materials (relevant documents). The aim of IR systems is to carry out a similar task automatically. For example, the user might issue the following request:

> *Have you got any documents pertaining to the Clinton Lewinsky scandal, particularly regarding his testimony before Congress?*

An automatic IR system then usually carries out some processing on the user request to derive a form of the request that it can match directly against the document collection using some form of *matching algorithm*. The processed request, which may take many forms, is known as the *query*. Query formats commonly employed in the IR world include the *natural language query*, where the request is not processed much at all, and the *bag of words* format, where function words, punctuation and phrases like "on the subject of" are removed from the request, suffixes stripped, and a selection of what are known as *keywords* extracted to form the query (see Figure 1.1). For example, the user request above, when processed in this manner, would yield the bag of words query:

> *clinton lewinsky scandal testimony congress*

Commonly used search engines, such as Google [2] ask the user to enter the bag of words query directly, thereby circumventing part of the request processing stage by getting the user to do some pre-processing in her head.

The document collection consists of a set of individual documents, each of which identifies a single text, such as a book, journal article, or web page. The documents in the collection can consist of the texts themselves, such as, for example, the document database of a web search engine, or of summaries that point the user toward the texts, such as book titles in a conventional electronic library catalogue.

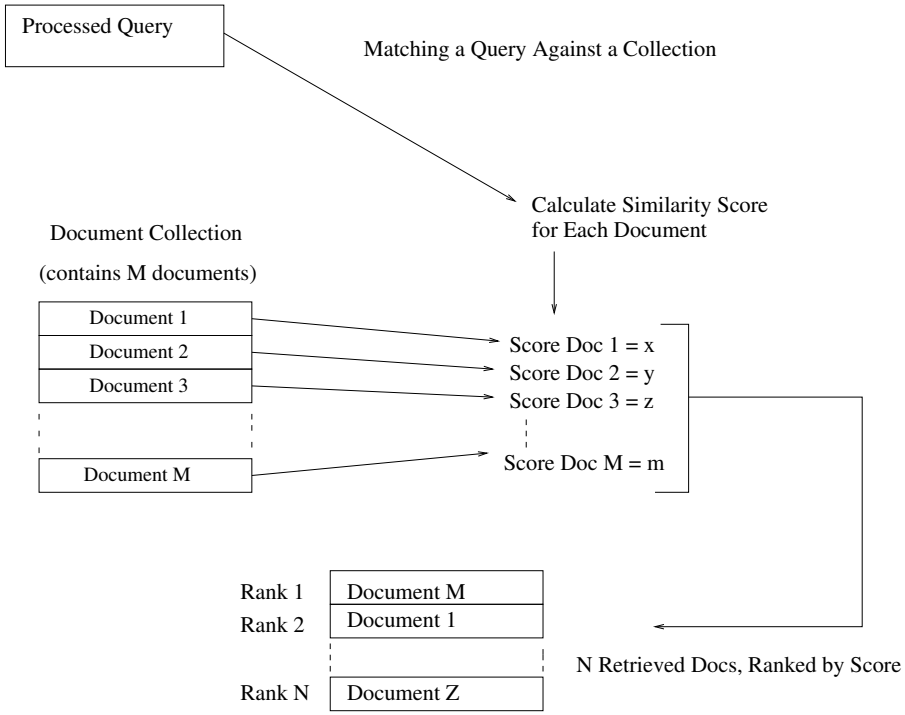

Figure 1.2: Matching a Query Against a Document Collection

The former case is known as *full text retrieval* and is the type of document collection we are interested in in our experiments. The document collection is also usually processed in an identical manner to user requests when it is being compiled. A common collection indexing strategy employed by systems that use bag of words queries is described in Van Rijsbergen 1979 [98].

The query is matched against the document collection using a *matching algorithm* which calculates a score for each document in the collection reflecting its perceived similarity to the query (see Figure 1.2). Similarity scores may be based simply on the frequency of individual query *terms* (words or phrases), or may exploit *term weights* (scores per term) calculated using frequency data. The Vector Space Method [87] and the Probabilistic Model of Information Retrieval [94] (which is the model employed by the IR system used in the experiments discussed in subsequent chapters) provide well-founded ways of doing this. Generally, a list of the N most closely matching documents is returned to the user. This list of returned documents is often called the *retrieved document list*. The aim is to retrieve as many *relevant* documents as possible in this list, while avoiding the retrieval of *irrelevant* ones. The notion of relevance is discussed further in chapter 2.

IR systems are generally evaluated using two metrics, *precision* and *recall*. *Precision* is defined as the proportion of retrieved documents which are actually relevant to the query derived from the user request.

$$Precision = \frac{NRel}{TotRetrieved}$$

Recall is the proportion of documents known to be relevant to the query in the entire collection that have been retrieved in the retrieved document list for that query.

$$Recall = \frac{NRelRetrieved}{TotKnownRel}$$

Various combinations of recall and precision have been employed as evaluation metrics in the literature. The more common composite metrics, including those used in our experiments, are presented in chapter 3.

1.3 Cross-Language Information Retrieval

Cross-Language Information Retrieval (CLIR) is where the user request and the document collection against which the request is to be matched are in two different human languages. The aim of CLIR is to match the request against the collection as if the request had been issued in the document collection language to begin with.

This kind of system is useful in the situation where a user who can read several different languages wants to find information in a collection containing documents in many languages, while avoiding the work involved in formulating multiple requests. For example, a Portuguese speaker, who also reads French, German and Spanish, might want to search a collection of EU directives for documents in these languages. Using a CLIR system, she could retrieve a list of, say, recent directives concerning the Common Agriculture Policy, in all of the dozen or so official EU languages with a single request. She then ignores retrieved documents in Swedish, Dutch and other languages she does not understand to concentrate on the documents returned that are written in one of the four languages she can read.

Research into CLIR is on-going, as the "perfect" system that would work in all situations and for all request-collection language pairs does not exist. The main problem in the field is selecting the appropriate translation of the important words or phrases in either the collection documents or the user request - in the latter case, without a great deal of contextual information being available. As we shall see in chapter 2, the option of using traditional Machine Translation (MT) techniques is not always available and alternative strategies have to be considered that are specific to CLIR. In particular, work on CLIR in a *resource-poor environment* (where large-scale expensive and scarce linguistic resources are not available due to the choice of language pair or subject domain) is far from complete. A summary of the state-of-the-art in current CLIR research is presented in chapter 2.

1.4 Why Dictionary-Based CLIR?

We shall see in chapter 2 that there are two main schools of thought with regard to CLIR. We can either translate the request, or the bag of words query derived from the request, at run-time into the language of each document collection in the system, or, conversely, translate each document collection into each language represented in the system at document indexing time. As the latter is cumbersome and time-consuming, and does not appear to offer any performance advantage over translating the request or query [71], most of the literature has concentrated on request or query translation, as have we.

The single most effective method explored to date is the use of conventional MT software for request translation [33, 37]. This method is entirely dependent on the availability of a suitable MT engine for the relevant request-document language pair. Commercial MT software is available currently only for a very small number of language pairs, and developing new systems or adding modules for new language pairs is an expensive and time-consuming process. Therefore, although one should certainly implement MT-based request translation if possible, we need to look to other approaches to request or query translation for the case where a suitable MT engine is not available to translate the requests.

Several studies have employed aligned bilingual corpora to extract translations automatically for CLIR, using these to translate requests, with some success [55]. However, these methods also depend on scarce resources, namely parallel bilingual corpora, which, although not as difficult to construct as an MT engine, nevertheless represent a significant financial stumbling block on the way to building a universally applicable CLIR system.

My research, and a great deal of current work discussed in the literature, concentrated on a particular type of request or query translation strategy which we call *Dictionary-Based CLIR* (DB-CLIR). This family of approaches includes all techniques which rely on a simple machine-readable bilingual dictionary to map the bag of words query derived from the user request to a semantically equivalent bag of words representation in the document language. DB-CLIR is important because although the CLIR strategies mentioned above which have been shown to be more effective [33, 55], DB-CLIR is the least resource intensive CLIR technique and thus the most widely applicable and extensively studied CLIR method.

DB-CLIR proceeds as follows: each *term* (semantic unit, can be a single word or a phrase, see chapter 2 for more on this subject) in the bag of words query derived from the user request is looked up in the machine-readable bilingual dictionary. Some form of ambiguity resolution or *equivalent selection* is applied to pick the "best" translation of that term from the list supplied from the dictionary. This "best" translation is then added to the document language semantic mapping of the bag of words query. This

document language semantic mapping is then matching against the document collection as if it had been directly derived from the initial user request.

1.4.1 The Four Steps of DB-CLIR

In chapter 2, we shall see that the process of obtaining a document language semantic mapping or *translation* of the bag of words query derived from the user request can be divided into four logical steps:

1. Pre-translation query modification
2. Dictionary lookup
3. Equivalent selection and weighting (this is the ambiguity resolution step)
4. Post-translation query translation modification.

(An *equivalent* is a candidate translation of a given query term).

We concentrated our efforts on the first three steps as existing work on the last step, post-translation query translation modification, was already felt to be sufficiently comprehensive. More specifically, we carried out most of our work on the dictionary lookup stage, where we examined dictionary characteristics with respect to the retrieval performance of resulting query translations. Very little attention has been paid in the literature to the lookup step or to the importance of dictionary characteristics, with most researchers preferring to jump straight into complex equivalent selection and weighting techniques instead.

In researching the effects of dictionary characteristics on retrieval performance, we were concerned with the situation where any additional resources required for the implementation of the complex equivalent selection and weighting methods discussed in the literature (see chapter 2) were not available, and where large-scale processing of the retrieval collection was not feasible - with the dictionary itself being the only accessible source of information. (It is not always be practical to carry out large-scale processing of the retrieval collection, such as that discussed by Ballesteros and Croft [10], if the collection changes daily or hourly, or if the collection is extremely large and the available computational power is limited).

We asked the question - **with careful examination of dictionary characteristics during the lookup step, along with some simple pre-translation query modification and post-lookup equivalent selection techniques which employ information found in the dictionary only, can we obtain a decent level of retrieval performance for associated query translations, without resorting to the more complex and involved techniques described in the literature? We found that the answer to the above question was YES**. Furthermore, a number of valuable insights into the process of DB-CLIR were made along the way.

We divide the discussion of our work below into four categories: our experiments on dictionary *scale* and *coverage*, our work on the *crucial equivalent effect*, our investigations of the effect on retrieval performance of the interaction between the type of retrieval system we used and *equivalent repetition within term's equivalent lists* in query translations, and the retrieval benefits of some simple ad-hoc ambiguity-based additional term weighting methods (by *additional* we mean in addition to and query terms weighting carried out by the matching algorithm employed by our retrieval engine). **Finally, we combined all of our insights to get a "best performing" set of query translations, which succeeded in nearly equalling the performance of CLIR using complex equivalent selection methods reported in the literature**, although a number of caveats apply when comparing IR systems and their results (see chapter 2).

1.5 Dictionary *Scale, Coverage Rate* and *Coverage Compensation*

We defined dictionary *scale* as the average number of distinct translations provided by our bilingual machine-readable dictionary for each query term. We found that the smaller the scale of the dictionary employed for query translation, the better the retrieval performance of associated query translations - provided a 100% *coverage rate* was provided. (The *coverage rate* is the percentage of query terms for which the dictionary can provide at least one candidate translation).

This result was due to what we termed the *swamping effect* of the larger number of irrelevant translations present in query translations obtained from larger scale dictionaries - many of these irrelevant

translations matched irrelevant documents, swamping any relevant documents present in the retrieved document list.

The best results were obtained by combining small-scale and large-scale dictionary entries in a process we called *coverage compensation* - this method ensured a high coverage rate by using information from a larger-scale dictionary whenever any query term did not have an entry in the smaller-scale dictionary being employed. (Smaller-scale dictionaries tend to be missing entries for more unusual words, and leaving out a "good" translation is highly detrimental to retrieval [47]). A small-scale dictionary has to provide a coverage rate of at least the 20% most ambiguous query terms it prior to the application of coverage compensation for this type of dictionary combination to be more effective than employing the larger-scale dictionary on its own.

Our work on dictionary scale and some of our work on coverage rate are presented in chapter 4, and the rest of our investigations into dictionary coverage rate in chapter 5. This work is described in two separate sections of two different chapters as our experiments are presented in chronological order.

1.6 The *Crucial Equivalent Effect*

The *crucial equivalent effect* is our name for the phenomenon of small differences in content between two translations of the same query resulting in big differences between them in terms of retrieval performance. The idea is that the omission of a single, *crucial* equivalent from a query translation can have a significant impact on retrieval performance. We looked at various ways of reducing this effect without increasing the swamping effect discussed above - some of our experiments were reasonably effective, some were not. Our work on the crucial equivalent effect is presented in chapters 4 and 5, again, in two separate places due to the chronological ordering of our experiments in this thesis.

1.7 *Equivalent Repetition* Within a Term's Equivalent List in Query Translations and Ambiguity-Based Additional Weighting Methods

Some dictionaries, especially those of larger scale, provide the same translation (or *equivalent*) more than once for a given query term, for example, for different senses of the original term. For example, the word *anxious* is translated as *anxiété* in two places in the Collins-Robert English-French Unabridged Dictionary [6] - once for the sense *troubled* and once for the sense *strongly desirous*. This can result in an equivalent being present more than once in the resulting query translation. The retrieval system we employed treated two different occurrences of the same equivalent within a query translation as two separate items, calculating a retrieval engine weight for both separately (see chapter 3 for the details of how the retrieval engine handles queries). In this case, as is the case with many other mainstream retrieval engines, repeating an equivalent is functionally the same as doubling its weight in the query translation. We found that such *equivalent repetition* could have a profound effect on the retrieval performance of query translations. In particular, we observed that less ambiguous equivalents (equivalents with fewer translations back into the source language) tended to benefit retrieval on being repeated, whereas more ambiguous equivalents did not.

This led us to conclude that the standard term weighting algorithm implemented by our retrieval engine could benefit from some adjustment in the cross-language setting, by increasing the weight calculated by the retrieval engine term weighting mechanism for less ambiguous equivalents, and decreasing that assigned to more ambiguous ones. Our work on equivalent repetition is presented in chapters 4 and 5.

1.8 Simple Equivalent Selection and Weighting Methods

Building on these conclusions, we investigated some simple equivalent selection weighting methods, where additional weights were applied to increase the importance of less ambiguous query terms and equivalents,

using ambiguity information from the dictionary. The use of this strategy improved the retrieval performance of associated query translations. Similar strategies were applied prior to query translation, with similar results. These weighting methods were ad-hoc in character and should be regarded as preliminary investigations into this type of equivalent and term weighting. This work is presented in chapter 6.

1.9 Overall Retrieval Performance

All of the experiments discussed above were designed to study the effect of individual factors, such as dictionary scale, on query translation performance in isolation. Therefore, the retrieval process was stripped down to the bare bones to ensure that no hidden factors were introducing artifacts into our results. This means that the absolute performance values reported in our experiments are rather low in comparison with the "best possible numbers" presented by others in, for example, the TREC proceedings [40]. Once the more advanced retrieval features implemented in the retrieval engine we used are reapplied and combined with our insights, we would get a more realistic idea of the kind of absolute performance our system could achieve.

Nevertheless, we found that adding simple additional weights based on ambiguity information from the dictionary to queries before translation and to query translations obtained using the best dictionary lookup methods discussed above resulted in a level of retrieval performance relative to an established monolingual upper bound almost as good as to the best in the field. This demonstrated that simple methods can take you a long way toward the goal of effective CLIR, without having to opt for complicated processing and/or resource mobilisation. This work is described in chapter 6.

Chapter 7 concludes this thesis with a summary of what has been achieved and some reflections on our work's wider implications within the field of CLIR, as well as suggesting some directions for future work.

1.10 Significance of This Work

Our work demonstrates that careful choice of a dictionary is crucial to success for dictionary-based CLIR. Existing research has preferred to gloss over the area of dictionary characteristics, preferring instead to go straight to complicated equivalent selection techniques. Therefore, we highlighted an hitherto neglected area of CLIR, showing it to be important for retrieval performance.

In addition, our work shows that it is not always necessary to use complex methods where simple methods will do - careful choice of dictionary is not only important, it can negate the need for complex methods. Furthermore, where it is not possible to implement complex equivalent selection strategies, for example, where the collection changes daily or hourly, it is still possible to construct an effective working CLIR system with a reasonable level of performance.

Finally, it means that a reasonably effective CLIR system can be implemented where minimal linguistic resources are available - bilingual dictionaries are available for almost every mainstream language pair you care to mention, and our methods use information from the dictionary only. This will aid in the dissemination of cheap, affordable, widely applicable CLIR technology.

The rest of this thesis describes current research and our experiments and findings in more detail.

Chapter 2

Background and Related Work

In the previous chapter, we introduced the concept of Cross-Language Information Retrieval (CLIR). Here, we present a review of the state-of-the-art in CLIR research, paying particular attention to those methods which are effective in a resource-poor environment or where large-scale processing of the retrieval collection is not practical. We also justify the avenues we chose to explore in the rest of this thesis with respect to current research.

We begin by explaining how the TREC conferences have played a key role in launching a new era of research for Information Retrieval, and describe how TREC led to the creation of two other cross-language evaluation initiatives, CLEF and NTCIR. A description of the methodology of TREC, CLEF and NTCIR follows, as many experiments reported in the literature were carried out as part of these initiatives or used a similar evaluation format, as did our own experiments.

Having reviewed the principal experimental methodologies, we note that there are three main families of approaches to CLIR - one may either translate the document collection into the language of the user request, translate the request, or a query derived from it into the language of the document collection, or transform both into some language-independent representation. Request or query translation methods (the most heavily investigated approach) can be further subdivided into three distinct categories: Machine Translation- (MT-) based, corpus-based and dictionary-based. (By *dictionary* we mean a machine-readable version of a simple bilingual dictionary, and not the more complex lexicon employed by many transfer-based MT systems).

These methods vary in terms of their effectiveness and their reliance on hand-crafted resources. The most effective CLIR method is to translate the user request using commercial MT software. However, this approach cannot be implemented when a suitable MT engine is not available for the given language pair. Document translation methods also rely extensively on MT software and suffer from the same limitation. Since developing an MT engine takes a great deal of time and effort, it follows that although commercial MT technology is very effective for CLIR we must consider other approaches to the problem. Corpus-based query translation methods do not rely on complex hand-crafted lexica like MT engines as they extract translation mappings automatically from a parallel corpus. Although such methods have been shown to be reasonably effective, they rely on the availability of a suitable parallel corpus for training purposes. Such corpora are scarce resources which have to be aligned by hand in the majority of cases. In addition, even though a corpus may be available for a given language pair, it may not be in the right subject area. As a result, we turn to dictionary-based translation of the bag of words query derived from the user request, as we still need a CLIR method for the case where MT- and corpus-based approaches cannot be employed.

Dictionary-based query translation systems are those which are the least reliant on expensive, scarce hand-crafted resources, as although the original dictionary does have to be hand-crafted by a lexicographic team, bilingual printed dictionaries already exist for a large number of language pairs. Furthermore, porting such a dictionary to a machine-readable format which can be utilised for CLIR, although non-trivial, requires considerably less human effort than any of the approaches to CLIR described above. This is why we have concentrated our research efforts on dictionary-based query translation, as we are interested in what can be done in the absence of scarce and costly hand-crafted resources.

For the purpose of clarity, we have broken the process of "translating" (mapping to an equivalent semantic representation in the target language) a bag of words query using a dictionary into four separate

steps: pre-translation query modification, dictionary lookup, equivalent (candidate translation) selection and weighting, and post-translation query modification. Most research has concentrated on the equivalent selection and weighting stage, as it is desirable to reduce the number of equivalents in the query translation much as possible, without removing any "correct" translations. Many of the selection methods presented in the literature are quite complicated, involving large-scale processing of the retrieval collection or of another linguistic resource. At present, the more sophisticated selection methods appear to lead the field by a reasonable margin in terms of retrieval performance. We pay particular attention to existing selection and weighting methods in this chapter as we shall be comparing their effect on retrieval performance with that of simpler approaches in our own research later on in this thesis.

The next section defines some terminology which we use in this and subsequent chapters.

2.1 Terminology

Here, we define some terminology which shall be used in the remainder of this document. The *source language* is the language of the user information request, and the *target language* the language of the document collection. A *query* refers to a user information request in the source language which has been converted to a format which can be directly submitted to a retrieval engine, usually consisting of a *bag* of individual words or *terms*. A *term* may consist of a single word, or of more than one word where a group of words in the query are held to represent a single concept, for example, *combine harvester*. Multi-word terms or *phrases* can be identified in the user request and placed in the query by traditional parsing technologies or by applying statistical methods [95].

By *translation* we do not mean a full translation of the user request incorporating correct grammar and word order as if it had been carried out by a human translator. Rather, we wish to perform a type of concept mapping, obtaining a target-language expression of the main concepts expressed in the source-language request, and are not interested in the detailed grammatical niceties which are so important in a traditional translation setting. This is firstly because the user request has had a great deal of grammatical information removed from it to derive the query and also because we do not seek to produce a readable high-quality translation of the request, but wish simply to match its constituent concepts (derived when the query is created) against the concepts contained in the target-language documents.

A translation of a query is never called a query here, it is always referred to as a *query translation*. A possible target language translation of a given query term is called an *equivalent*. We cannot guarantee the identification of a unique equivalent for every term, for example, a given English term may have several possible French translations. Each of these possible translations is a French-language equivalent of the English term, and so we allow each source-language query term to be associated with several equivalents.

Blind relevance feedback or *pseudo-relevance feedback* is a process whereby an initial retrieval run of the query translations is carried out using either the retrieval collection or a separate document collection employed for the purposes of performing relevance feedback only. In traditional relevance feedback, the user indicates which of the the top N retrieved documents are relevant to the query and only these are retained for the next step. In *blind* relevance feedback, all of the top N documents are assumed to be relevant, and thus all are retained. There is no need for any user involvement. The terms in the retained documents are then ranked in order of desirability of inclusion in a new version of the query. Lee [59] presents several commonly employed ranking formulae. The most common form of blind relevance feedback is to select the top M terms from the ranked list of terms and add them to the query. New weights are calculated for both these new query terms and the existing query terms and this revised query run again. There are also feedback methods which delete terms from the query, or which recalculate weights for the existing query terms only, without adding any new terms.

Several iterations of this process are possible; in practice, one suffices. Once the feedback process has been completed, the resulting revised query is run on the retrieval collection. (In the case where documents from the retrieval collection itself were used to generate the new query, these are generally not included in the collection for this run). This process can also be applied to the source-language query before translation, using a document collection in the source language (see section 2.12). Blind relevance feedback has been shown to be effective in a monolingual situation by many researchers [20].

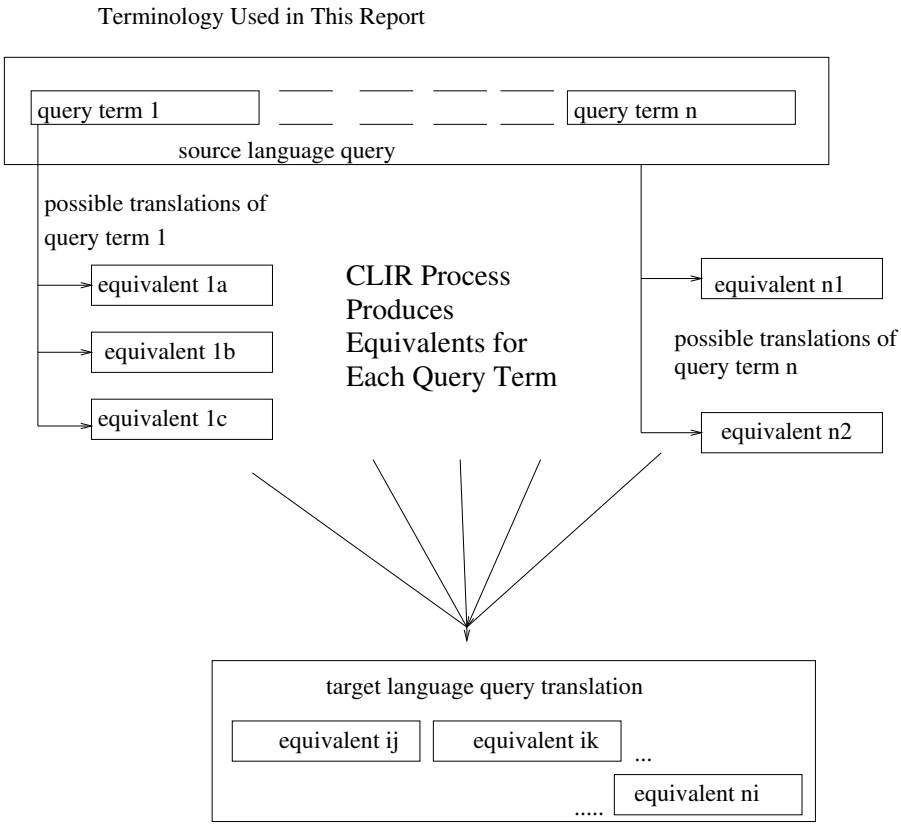

Figure 2.1: Terminology Defined in This Thesis

2.2 The Goal of Modern CLIR Research

The first CLIR experiment was carried out by Salton in 1973 at Cornell University [85]. Salton constructed a multilingual thesaurus by adding appropriate German translations of English concepts to his retrieval system by hand. This thesaurus was then looked up at run-time to obtain translations for individual query terms. The German translations which were added to the thesaurus were selected specially to translate the test queries obtained from the test set of user requests accurately. It was discovered that performance levels similar to that recorded for monolingual IR could be achieved in this manner. The challenge of modern CLIR is to achieve a similar level of performance while carrying out any translation or transformation of the request, query or document collection automatically and without tailoring the system to a particular set of requests.

2.2.1 TREC

Interest in IR generally and in CLIR particularly began to grow with the advent of NIST's TREC (Text REtrieval Conference) initiative in 1992. Harman noted in the overview of the first TREC that the two main problems affecting IR research at the time were the difficulty in building on others' research due to the lack of a standard collection, request set and evaluation data, and a dearth of real-world-sized full text retrieval collections complete with evaluation and relevance data available to researchers [42]. TREC attempted to address both of these problems by providing a framework for retrieval system evaluation and comparison using a large-scale real-world full-text document collection.

The first TREC comprised two separate retrieval tasks, which were to be carried out on a large (initially 2GB) heterogeneous document collection. Here we examine the *ad hoc* or general retrieval task, which consisted of retrieving documents from this collection for a set of *topics* (the other, *filtering* task is not really relevant to the work discussed in this thesis). A topic is a structured expression of a user information need from which requests and then queries may be derived. Participants in TREC-1 were issued with 50 topics and the document collection, and sent lists of the top 1000 documents retrieved by their IR system for each topic back to NIST.

At TREC-1, NIST evaluated each system for each topic using a technique known as the *pooling mechanism*. There were too many documents in the TREC collection to allow each to be judged (deemed relevant or irrelevant) for every topic, and so, for each topic, the top 200 documents returned by each participant for a given topic were pooled and then each document in this pool assessed as either relevant or irrelevant to the topic in question by a team of human experts. This set of relevance judgements was the *pool of relevance judgements* for that topic. The list of 1000 documents retrieved by a given participant's system for a given topic was then evaluated with respect to that topic's pool. The assumption was made that nearly all relevant documents would have entered the pool, thus documents which appeared in one of the lists of top 1000 documents but not in the pool for a given topic, and which had therefore not been judged, could be assumed to be irrelevant.

Performance for each topic for each participant was calculated using a number of metrics [42], including Average Precision (hereafter known as *AvP*) and Exact Precision (*R-Prec*). AvP is an average of precision values calculated at every position in the retrieved document list where a relevant document is encountered. It most frequently calculated as a single figure, or displayed in a table showing AvP values at 11 points of recall ranging from 0.0 to 1.0. AvP at a level of recall of 0.2, for example, is the precision at the point in the ranked retrieved document list where 20% of the known relevant documents have been retrieved. R-Prec is precision taken at the single point in the document list where precision is equal to recall. Values at a number of fixed points in the retrieved document list are also provided in a table, for example, at 10 or at 100 documents. Overall AvP for a set of topics is the single most quoted evaluation metric in the literature as a measure of system performance. We have used the same metrics to evaluate retrieval performance in our own experiments - using a program called `trec_eval` developed by NIST and distributed to TREC participants and the public at large for the express purpose of calculating these metrics for retrieved document lists presented in the approved TREC format.

In subsequent TRECs the same evaluation methodology was employed, but with the pool depth being reduced to 100 documents and new sets of topics being issued every year. Although the TREC definition of relevance may be viewed as somewhat artificial, it is considered to be a good basis for comparing one system with another and is widely accepted as an accurate reflection of comparative performance [99]. TREC-01 (the 10th TREC) took place in November 2001. Numerous new tasks or *tracks* have been added

over the years, such as question-answering, high precision, interactive, Chinese, and cross-language. The ad hoc task was discontinued in 1999 and replaced with the Web Retrieval track. New sets of topics are issued each year for each track which is being run.

A multilingual track was introduced at TREC-4 [41], where participants applied what they knew about retrieving English-language documents to a Spanish collection. This track was run again at TREC-5 the following year, as was a similar Chinese track. Two sets of participants, Davis and Dunning at NMSU, and Ballesteros and Croft at the University of Massachusetts, Amherst, performed some cross-language work as part of their Spanish retrieval experiments [26, 8]. A CLIR track, covering English, French and German, was introduced at TREC-6 in 1997 [89]. Participants matched requests in one language against a document collection in one other. At TREC-7, the task was changed to matching a request in one language against all the CLIR document collections, retrieving a multilingual list of documents. Italian was added at TREC-8 [14, 12].

We shall pay close attention to the methodology of the TREC CLIR task here as many of the experiments reported in the literature were carried out as part of the CLIR track of TREC-6, TREC-7 and TREC-8. In addition, numerous researchers, including some who were not participants in TREC itself, have used the TREC CLIR data to perform their own retrieval experiments, many of which are also described here. Finally our own experiments have utilised the TREC CLIR English and French language data.

The AP (Associated Press) newswire documents, part of the original TREC ad hoc collection, were the designated English-language collection for the CLIR track at TRECs 6, 7 and 8, numbering around 240,000 documents and approximately 750MB in size. This collection consisted of short news stories, ranging from the 1st of January 1988 to the 31st of December 1990. Similar collections in French and German of size 250MB and 330MB containing approximately 140,000 and 185,000 documents respectively were provided by the SDA (Schweizerische Depeschen Agentur - Swiss News Agency) newswire, for the same range of dates. An additional collection was available in German, the NZZ (Neue Züricher Zeitung) collection of size 200MB, consisting of newspaper reports from 1994 and containing c. 67,000 documents. The TREC-8 Italian collection was also provided by the SDA for the same date range as the other SDA collections. It was around 150MB in size and contained approximately 63,000 documents.

For TREC-6, NIST provided 25 topics, each of which was available in English, French and German. Evaluation for TREC-6 was carried out using the pooling mechanism described above [89]. The 28 topics for TREC-7 were created as follows. For each of the four languages in the evaluation, seven topics were created at a site where the language in question was the native language. Each of these 28 topics were then translated by hand into each of the other three languages in the evaluation. This meant that a human-generated version of all 28 topics was available in all four languages. The pooling and evaluation part of the TREC task was also distributed over several sites. A similar methodology was implemented for the TREC-8 CLIR track. The test query set employed in all of our experiments were derived from these 81 topics from TRECs 6 to 8.

Evaluation for the CLIR track at TRECs 6 to 8 proceeded in a similar manner to that carried out for the ad hoc task, with a pool depth of 100 documents. Voorhees points out that there are some additional problems with obtaining relevance judgements for cross-language collections [99]. Firstly, the relevance judgements are harder to obtain, as this involves multiple assessors for each topic (one per language), and secondly, ensuring a large and diverse pool for each topic and each language is more difficult to co-ordinate. Finally, results are not balanced across languages - there may be considerably more relevant documents for a given topic in, for example, the English collection than in the German one. However, even when different assessors are employed, thus resulting in different sets of relevance judgements, the relative performance differences between systems are maintained, and so the TREC pooling mechanism for relevance assessment may be viewed as reasonably reliable in a cross-language setting [99, 90].

2.2.2 CLEF

In 1999, it was decided to discontinue the European language CLIR track at TREC, replacing it with a CLIR track concerning English to Mandarin Chinese retrieval [36]. At TREC-01, the CLIR track concerned itself with English-Arabic retrieval. The European language CLIR evaluation has continued under the auspices of the Cross Language Evaluation Forum (CLEF) [74]. Following the tradition of the TREC-8 CLIR task, CLEF has incorporated an increasing number of languages, not all of them European. For example, at CLEF 2001, 50 topics were made available in Dutch, English, French, German, Italian,

Spanish, Mandarin Chinese, Finnish, Japanese, Russian, Swedish and Thai, with 33 participants. The pool depth for CLEF has been set at 50 documents.

Evaluation has proceeded as for the TREC-8 CLIR task. CLEF did not retain the TREC-8 document collections, opting instead for collections of newspaper texts from a given time period in German (Der Spiegel), English (Los Angeles Times), French (Le Monde) and Italian (La Stampa). Additional collections in Dutch and Spanish were added for CLEF 2001. There were three separate tasks at CLEF 2000 - monolingual, where retrieval methods known to be effective in one language were tested on others, bilingual, consisting of topics in one language retrieving documents from a collection in one other (like the TREC-6 CLIR task), and multilingual, where topics in one language were to retrieve documents in all languages represented in the CLEF collections (similar to the TREC-7 and TREC-8 CLIR task).

2.2.3 NTCIR

Also in 1999, NACSIS in Japan launched a TREC-like initiative for Japanese IR and English-Japanese CLIR called NTCIR. The NTCIR document collection is more than 300MB in size and contains approximately 330,000 documents in Japanese, consisting of technical abstracts [73]. The topics are similar in structure to TREC topics and evaluation is performed in an identical manner to TREC, employing the same evaluation software. NTCIR is planned to continue on an annual basis.

2.2.4 TREC Topics

As both CLEF and NTCIR have continued to employ the topic format defined initially at TREC, and our own experiments use this topic format, we provide here a detailed description of topic structure. A TREC topic expresses a user information need and consists of three main fields, the *title, description* and *narrative*. The title is a two or three word summary of the general subject area covered by the topic. The description field contains a one-sentence account of the user's information need. Finally, the narrative comprises a short paragraph describing this information need in more detail. The narrative was originally included to aid NIST assessors in determining relevance. Although most experimenters create a user request set by extracting the topic description field or the title and description fields together, the narrative field has been included in the derived requests in some CLIR experiments. Figure 2.2 shows the contents of these three fields for TREC-6 cross-language topic 1.

Frequently, the extracted requests are then processed to derive bag of words queries. The types of processing applied include stemming or lemmatisation, the removal of stopwords and punctuation and the conversion of upper-case letters to lower-case (a more detailed discussion of the relative benefits stemming and lemmatisation of the topics and document collection may be found later on in this chapter). Stemming is the removal of suffixes [98] whereas lemmatisation is the conversion of an inflected word to its uninflected dictionary form [53].

In this manner one derives an unordered collection of terms (bag of words), which is submitted to a standard IR engine as a query. For example, by extracting and then processing the description field of cross-language topic 1, one might obtain the query:

```
reason controversy surround waldheim world war ii action
```

The bag of words approach is not employed by every CLIR method described in this chapter, although it is the most widely implemented form of query and is that used in our experiments.

2.3 Performance Issues in CLIR

Salton demonstrated that performance for CLIR could be the same as for monolingual IR providing any necessary translation was carried out accurately [85]. However, modern CLIR systems do not yet perform at that level [17] - Salton performed his translation manually, whereas modern systems attempt to do so automatically. Hull and Grefenstette cite *translation ambiguity* - the difficulty of choosing the right translation for a given term in a bag of words query - as one of the main problems in automatic CLIR, as well as the failure of the system to find a translation for every term (known as *lack of coverage*) and the incorrect translation of *phrases* [47].

Field	Content
Title	Waldheim Affair
Description	Reasons for controversy surrounding Waldheim's World War II actions.
Narrative	Revelations about Austrian President Kurt Waldheim's participation in Nazi crimes during World War II are argued on both sides. Relevant documents are those that express doubts about the truth of these revelations. Documents that just discuss the affair are not relevant.

Figure 2.2: Contents of Fields in Cross-Language Topic 1

We noted above that a query term could consist of more than one word. Such *multi-word terms* or *phrases* may be detected in documents, requests and/or queries by applying either traditional parsing technology or statistical term grouping techniques [95]. Finding phrases and translating them correctly is the subject of considerable research activity [9, 5]. Since phrases tend to have fewer equivalents in the target language, this reduces translation ambiguity and therefore aids performance. As such, one may view phrases as being normal bag of words query terms with fewer target-language equivalents. We have chosen to focus on single-word terms only in our research, because we wish to study aspects of the mechanisms of translation in isolation instead of trying to obtain the highest possible retrieval performance scores, and therefore phrase recognition is incidental to our experiments.

Some modern CLIR research incorporates the merging of lists of retrieved documents from a number of different collections [16]. We do not address this problem as this is a separate area of research which has already been investigated in monolingual IR research [39]. Our interest is limited to the translation process.

2.4 Evaluation Issues

There are two separate factors which combine to produce the overall retrieval performance score registered during a CLIR experiment. The first is the effectiveness of the CLIR method employed, the other is the quality of the retrieval engine. This means that it is difficult to make a direct comparison between two sets of results from, for example, the same TREC CLIR evaluation, as it can be difficult to disentangle the performance gains which are due to the CLIR method from those which result from the use of a certain retrieval engine.

Therefore, wherever possible, we express performance of a CLIR run as a percentage of the performance of the *corresponding monolingual run*. The corresponding monolingual run is defined as running an accurate human-translated version of the request set on the target-language document collection. Since participants in the evaluations discussed above were supplied with correct versions of each topic in each language under consideration, it was possible for them to perform such runs. Where monolingual results were not provided, we have quoted the absolute AvP and R-Prec values for the cross-language runs under consideration. These must be treated with caution as there is more than one factor influencing the results.

There is also a data effect when we enter the realm of CLIR with many languages, such as in the TREC-8 CLIR track, where documents in many different languages are to be matched against a query or request in yet another. The distribution of relevant documents across languages in the document collection is rarely uniform. For example, if we have a collection of Spanish and Italian documents being matched against queries in French, if most of the relevant documents are contained in the Spanish sub-collection, performance is likely to be better for French-Spanish retrieval than for French-Italian, irrespective of the quality of the CLIR strategy employed for the latter language pair.

As such, it is very difficult to compare one system's results with another. Thus it may not be possible to do more than to note general trends. Although we do sometimes quote numerical results where the corresponding monolingual results are not available, extreme caution must be exercised when interpreting them. Even for single collections and experiments, results vary widely as illustrated in figure 2.3.

Participant	TREC-7	TREC-8
CLARITECH	-	24
Eurospider Information Technology AG	28	19
IBM	32	26
IRIT/SIG	-	21
Johns Hopkins University APL	-	26
New Mexico State University	-	15
Twenty-One	30	25
University of California, Berkeley	24	-
University of Maryland	16	16

Figure 2.3: AvP of Best Run for Participants in the TREC-7 and TREC-8 CLIR Tracks

2.5 Approaches to CLIR

We may divide current CLIR research into three categories:

- Document Translation
- Query/Request Translation
- Non-Translation Based Methods

We shall see that document translation is expensive and time-consuming, and that the approaches which are not based on some form of direct translation are somewhat limited in application. As regards query or request translation, the best-performing method is the use of commercial MT software to translate the user request directly without processing it to create a bag of words query, but it can only be applied where a suitable MT engine is available for the relevant language pair. Using corpus-based methods to translate the bag of words query derived from the user request are an alternative, yielding reasonable retrieval performance scores, but are limited to subject domains and language pairs for which a suitable parallel corpus is available. Dictionary-based translation of queries, although not always the best performing method, can be applied to any language pair for which a simple machine-readable bilingual dictionary can be located. We have identified the main problem in current CLIR research to be determining what to do when hand-crafted resources, such as MT engines and bilingual corpora, suitable to the task in hand cannot be obtained. Therefore, we have chosen to focus on dictionary-based query translation in our research, as it is the method which relies the least on expensive, scarce resources.

2.6 Document Translation

Document translation comprises approaches to CLIR which require that all documents in the collection be translated into the language of the original user request. User requests or derived queries are then dealt with by a monolingual IR system.

The principal translation method reported in the literature is commercial off-the-shelf MT. The rationale behind this approach is that whereas user requests are often viewed as being too short to provide sufficient contextual information for traditional MT to perform well, documents may be translated as normal texts. This approach may be implemented without developing any CLIR-specific software. Nothing is needed other than a commercial MT product and a standard retrieval engine. Problems such as translation ambiguity and coverage are dealt with in a "black box" manner by the MT software.

Oard and Dorr at the University of Maryland translated the German TREC-6 CLIR document collection into English using Logos, a commercially-available MT engine [71]. Retrieval was then performed running bag of words queries derived from the TREC-6 English topics on the translated collection using the INQUERY retrieval engine [18]. AvP of 53% and 79% of corresponding monolingual German retrieval was recorded. These results were similar to those obtained for request MT employing the same data and retrieval engine.

There are several problems with document translation for CLIR. The first is that the cost in terms of time and money of translating a large document collection using traditional MT technology can be prohibitive. It took Oard and Dorr two months to translate the 550MB TREC German collection [71].

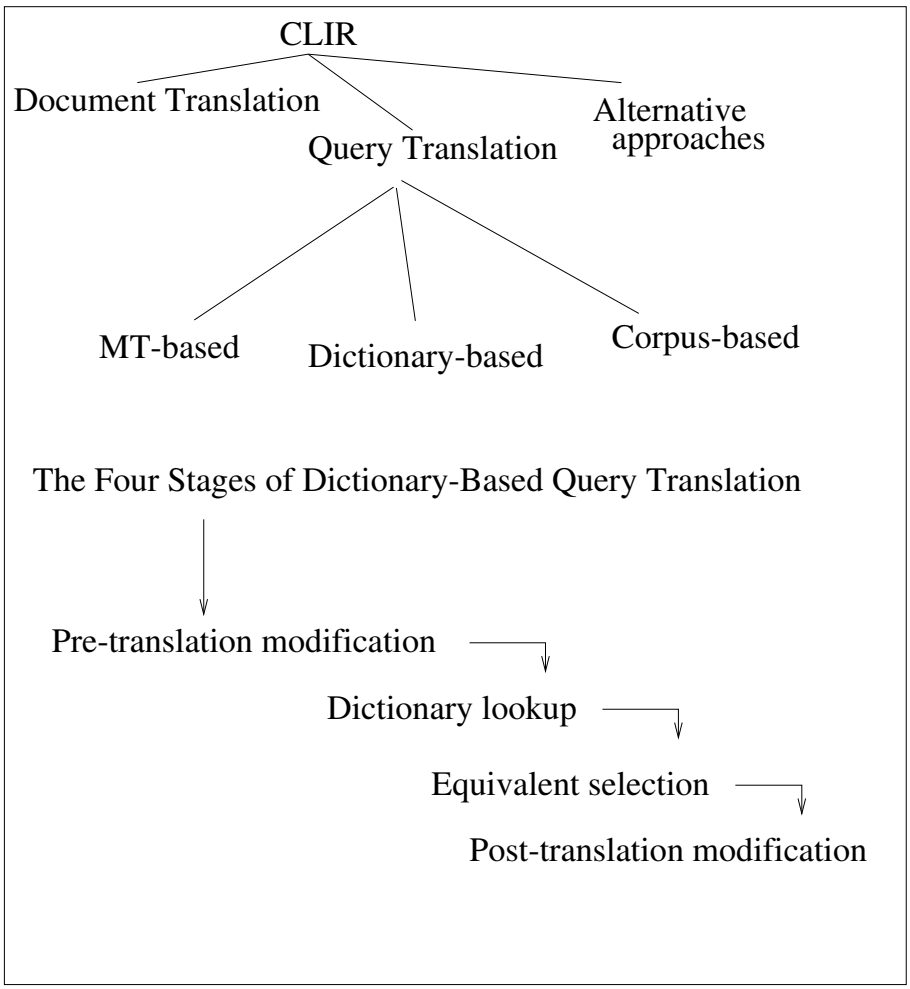

Figure 2.4: Approaches to CLIR Described in the Literature

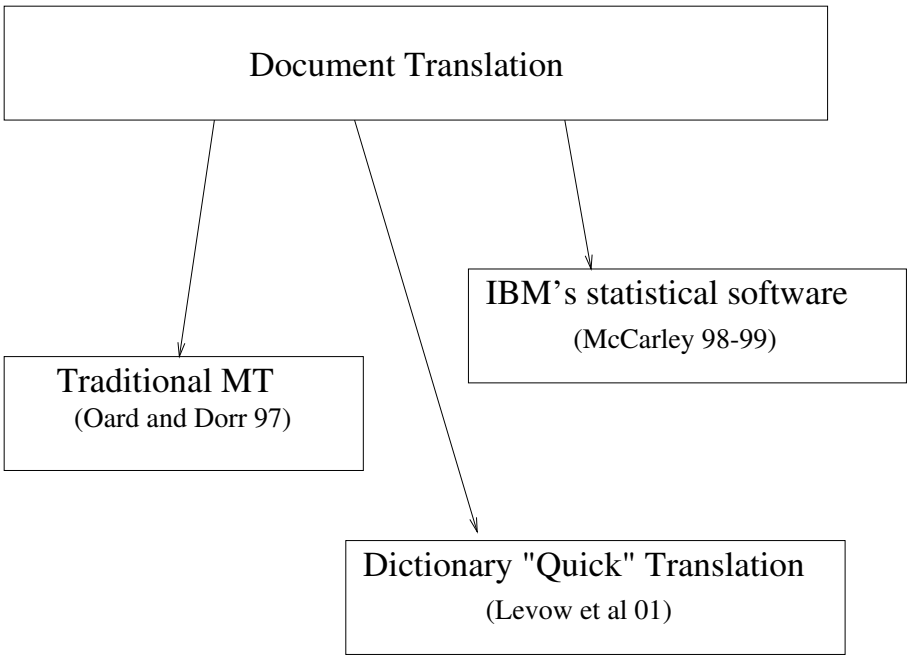

Figure 2.5: Document Translation Techniques

Although computing power has decreased in cost a great deal since 1997, document collections have grown larger. This is all the more the case where more than two languages are represented in the system - each document would need to be translated into each language represented in the collection and also into each possible request language. In addition, the development of a high-quality and comprehensive MT system is a slow and expensive process which often takes years to complete. The same is true of porting existing MT software to a new language pair. Commercial MT software is available currently for very few language pairs.

Researchers at IBM's TJ Watson Research Centre attempted to reduce the time needed to translate a document collection by applying their statistical machine translation software to document translation for TREC-7 [33]. Their system used an English-French parallel corpus (transcripts of the proceedings of the Canadian parliament) as training material to build a statistical language model [19]. Translation of the French SDA collection to English using this model was accomplished fairly rapidly. Average precision of 34% was recorded for translating the French TREC-7 collection into English and then running queries derived from the English TREC-7 topics on this translated collection. No corresponding monolingual retrieval performance value was supplied.

To get around the lack of suitable parallel corpora for the other language pairs in the TREC CLIR task, IBM also explored the use of story-level alignment of the SDA collection for French and German. These collections contain many documents which are similar in content (for example, both collections contain reports on the fall of the Berlin Wall in 1989). Meta-information supplied with the SDA documents allows documents with equivalent content to be aligned at the story level. This subset of aligned stories was then used as a French-German *comparable* corpus to train the statistical software in the same manner as a parallel corpus. (A comparable corpus is a bilingual corpus which is not parallel, but where both parts are of similar content). This method was also employed to produce a language model for German to Italian translation.

To translate from English to German, French was used as a pivot language. This means that the English documents were translated from English to French, and then from French to German.

Unfortunately, the results for retrieval based on using comparable corpora to train the statistical translation software were not nearly as good as when a parallel corpus was employed. For example, AvP of 24% was recorded for retrieval following the translation of the French collection to German, compared to 34% for French to English translation as reported above. Performance dropped even further where a pivot language was used, various results between 13% and 22% AvP are quoted for English to Italian

retrieval. It was clear that to obtain reasonable retrieval performance using IBM's software, a suitable parallel corpus must be available for the language pair selected. This severely limits the applicability of IBM's software - see section 2.10 for more on the limitations of (simpler) parallel-corpus-based methods.

Oard's team also attempted recently to translate documents using Select-First-N, n=2 simple dictionary lookup [60] (see section 2.14.3 for details of this method). The document collection was processed as if it were being prepared for traditional indexing, with, for example, the removal of stopwords. Then, documents were translated by replacing each term in each document with its first two equivalents from a machine-readable bilingual dictionary. Where a term had only one equivalent listed for it in its corresponding dictionary entry, this equivalent was included twice in the translation of the document, thereby ensuring that two equivalents were provided for every source-language document term. The document collection could be "translated" rapidly using this method. Retrieval was performed using all three main topic fields (title, description and narrative) to derive the queries. Results between 24% and 30% AvP were obtained.

Braschler *et al* at Eurospider Information Technology AG found document translation to be most effective when combined with MT-based request translation and their similarity thesaurus method (see section 2.10.3) for the CLEF 2001 multilingual task [15].

Document translation using traditional MT suffers from a lack of available systems for many language pairs. IBM's software, on the other hand, is limited in a similar manner by the lack of available parallel corpora. Oard's dictionary-based approach needs to be investigated further before any direct comparisons can be made. Finally, where MT technology is available, it shall be shown below that request MT can perform just as well, without the associated expense.

2.7 Non-Translation-Based Methods

There is a small number of approaches to CLIR which translate neither the requests/queries nor the documents, opting instead to convert both to a language-independent representation where they can be searched directly [46, 79].

The only such system to be entered in a large-scale evaluation such as TREC was the CINDOR system at Textwise Corporation which used Wordnet *synsets* [31] as a multilingual thesaurus to mediate between requests and documents [30]. However, despite the existence of projects like EuroWordNet which aim to translate Wordnet into languages other than English by hand [101], Wordnet is still limited in its coverage, and it is difficult to see how it could be expanded without considerable work. Considerable improvements in performance were recorded at TREC-8 for the CINDOR system by switching to using MT software for request translation [61].

2.8 Request or Query Translation

We have seen that it is not usually feasible to translate each document in the collection into every language represented in it, and that existing techniques which map both documents and queries or requests to an interlingua representation require as much hand-crafted knowledge as document MT but do not perform as well. In this section we examine the obvious alternative - translating the requests or queries into the languages of the document collection.

There are three main query translation methods:

- *Request MT.* This is where a commercially-available MT engine is used as a "black box" to translate the user request as-is.
- *Corpus-Based Query Translation.* This is where techniques from the domain of corpus linguistics are applied to map the terms in the bag of words query derived from the user request to a semantically equivalent representation in the target language.
- *Dictionary-based Query Translation.* This is where a simple machine-readable bilingual dictionary is employed to map the terms in the bag of words query to an equivalent representation in the target language.

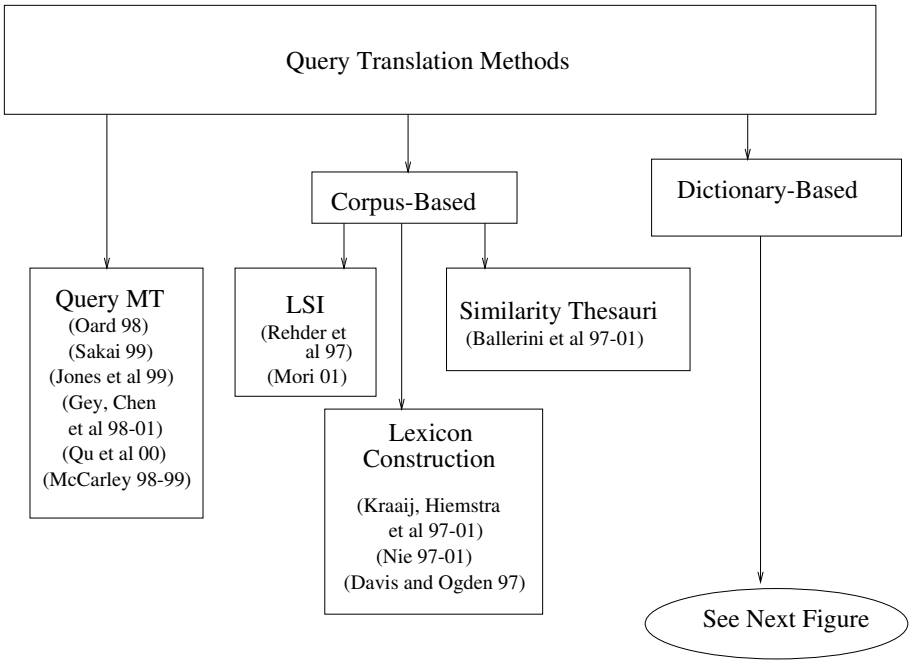

Figure 2.6: Approaches to Query/Request Translation

A detailed account of current research in each of these three areas follows, including not only the relevant TREC, CLEF and NTCIR experiments, but also work carried out using data from these initiatives but not actually part of the evaluation exercise.

2.9 Request MT

We saw that document translation, although reasonably effective, was often not feasible because of the time and expense associated with translating every document into every language represented in the system. This situation can be avoided by using commercial MT software as a "black box" to translate the user requests instead. For TREC experiments, this usually involves the translation of the single sentence contained in the topic description field, although sometimes all three main topic fields, the title, description and narrative, are concatenated in a single paragraph of text to create the request passed to the MT engine. The monolingual IR system employed may or may not process the resulting target language request to derive a bag of words query - this level of detail is not usually provided in the literature, however, the bag of words approach is the most commonly employed query format.

Another advantage of request MT is that, as stated in section 2.6, no special software is needed other than the commercial MT package used and a standard monolingual IR system. In addition, it is not nearly as expensive in terms of time or resources to translate a relatively small set of requests as to translate an entire document collection. Furthermore, the lack of contextual information available to the MT engine when the shorter TREC topic fields only are used to create requests does not seem to be a problem - as we shall see below, request translation using MT has performed quite well in many cases, for example, many of the top scoring participants in the TREC-7 CLIR track employed some form of MT [33, 37]. In addition, comparative evaluation using the same data and retrieval engine have shown request MT to be as effective as document MT and superior to dictionary-based approaches [70]. Therefore, we would suggest that request MT should be the CLIR method of choice where a suitable MT system is available. However, the availability of MT systems for new language pairs, as discussed in section 2.9, remains a serious problem.

At TREC-6, Oard and Dorr translated the German requests derived from the TREC-6 CLIR topics into English using Logos [71] and ran them on the TREC CLIR English collection. Two types of requests were constructed - *short* requests, comprising the title field only, and *long* requests, consisting of the

title, description and narrative fields together. The University of Massachusetts' INQUERY retrieval engine was employed to perform these experiments. AvP comprising 64% of monolingual performance was obtained for the long requests, and 68% for the short requests, which compares favourably with the results for document translation obtained in the same set of experiments and discussed in section 2.6.

In another experiment, Oard and Dorr compared dictionary-based bag of words query translation with their MT approach, obtaining very similar results for CLIR for the short requests [69], again using INQUERY. The MT translations of the long requests performed better than the corresponding dictionary-based translations. At TREC-7 results were similar, with an AvP reaching 64% of monolingual performance for English to German request translation and the German collection [70], this time using NIST's PRISE retrieval engine [77]. Sakai also experimented with MT for request translation at Toshiba [84, 82]. The NTCIR Japanese topic description fields were translated to English using Toshiba's ASTRANSAC MT engine [84], and two different retrieval engines employed, one based on the probabilistic model [94], and one utilising the vector-space model. These translations obtained 72% of the average precision of English monolingual retrieval. Sakai also verified these conclusions using a different Japanese to English MT engine, Toshiba's MTAvenue system, and the TREC collections [82].

Jones *et al* found MT to be superior to a number of dictionary-based translation methods [50] using the same probabilistic IR engine as Sakai. These were: selecting the first available equivalent for each query term (see section 2.14.3), using all synonyms of that first available equivalent as the translation of that term (synonym generation was carried out by the MT system), and including in the query translation those equivalents whose part of speech matched that of the original source-language query term only (see section 2.14.4). Several other experiments reported in the literature have also shown MT-based request translation to be quite effective for CLIR [37, 78].

IBM reported results for both document translation and request translation, obtained using their statistical machine translation software and the TREC-8 data (see section 2.6) and a proprietary retrieval engine based on the probabilistic model, but no direct comparison with other forms of request or query translation were performed [63].

Despite its reliance on parallel corpora, we classify IBM's system as a full MT package because it aims to tackles all aspects of translation, such as alignments between sentences in the source and target languages and source language fertility. . The corpus-based query mapping methods discussed in the next section do not try to perform full MT, seeking merely to map the concepts expressed in the bag of words query derived from the user request to equivalent concepts in the target language.

Gey *et al* at the University of California, Berkeley, compared three commercial MT systems, obtaining similar retrieval performance for request translation for all three [37] for the TREC-7 CLIR task. Retrieval performance for translating English-language request into each of French, German and Italian was good, with AvP of 34% being recorded. No corresponding monolingual benchmark was provided. As the software available was only able to translate between English and other languages, English was used as a pivot language to translate between French, German and Italian. Performance for these runs was somewhat inferior to that mentioned above, ranging from 21% to 24%. This demonstrates once more that a pivot language approach is not a satisfactory solution. The same group also discovered that employing a domain-specific bilingual lexicon (see section 2.10.1) to translate requests or queries could result in better retrieval performance than using MT [21] at NTCIR-1 in 1999. However, the retrieval collection employed in these runs consisted of abstracts of technical papers rather than general texts. Further experiments on less technical data would be necessary before any conclusions regarding retrieval in general could be drawn.

One would expect performance (measured in terms of AvP) to be highly dependent on the quality of the translation produced by the MT engine. Sakai discovered that this was not necessarily the case, provided a blind relevance feedback step was applied after translation [82].

Sakai discovered that applying a post-translation blind relevance feedback step could compensate for poor MT for many requests [82]. When he applied blind feedback to the queries derived from his request translations, performance jumped from 77% to 97% of monolingual retrieval. Qu's results corroborated these findings [78].

Others have tried combining more than one MT system in order to compensate for gaps in any one system's coverage. Jones and Lam-Adesina at Exeter University found that combining two MT systems for the CLEF 2001 multilingual task resulted in slightly better results than using either MT engine alone, although performance was not as good as when a single MT engine was employed and the queries derived from the request translations expanded using a document summarisation technique [49]. Braschler *et al*

at Eurospider Information Technology AG found that performance improved when MT was combined with other methods [15], as did Savoy at the University of Neuchâtel [88] and Gey *et al* at the University of California, Berkeley [38]. This suggests that combining the output of multiple methods, particularly multiple MT engines, in request or query translation could be beneficial for retrieval performance.

We concluded that where MT software was available for a given language pair, it should be used for request translation. In addition, combining the output of several engines, augmenting the MT output with the output of some of the less effective methods discussed below and implementing a blind relevance feedback step would all constitute desirable features of a working CLIR system. However, the relevant technology is not always available, and so we need to consider other approaches to the problem of request or query translation for these cases.

2.10 Corpus-Based Query Translation

Corpus-based techniques for query translation or mapping involve a training phase where a bilingual parallel corpus in the relevant language pair is employed to create mappings or *correspondences* of source-language terms to
target-language equivalents. These correspondences are then used to map the terms in the bag of words query to equivalent terms in the target language. In this way these approaches differ from IBMs full statistical MT strategy, which tackles the MT task in its entirety.

Selection of the correct translation equivalent for a given term in a query derived from a user request is performed implicitly by the statistical model developed during the training phase, and coverage depends on the breadth of subject matter and content of the training corpus. This means that the translation information does not need to be hand-crafted, but is extracted automatically from the parallel corpus, thus reducing the amount of hand-crafted knowledge needed to perform translation.

However, since the methods presented here need a suitable parallel corpus for training purposes, this constitutes an important obstacle for their implementation in a CLIR system, as sentence-aligned parallel corpora need to be aligned by hand. (Research into the automatic compilation and alignment of parallel corpora is ongoing [92]). Since aligning a large corpus by hand is time-consuming, such corpora are scarce and expensive to construct. In addition, it is not clear that a system trained on a parallel corpus in one domain could be readily applied to translate queries in another. If performance drops fairly rapidly as the retrieval collection and parallel corpus diverge, this would severely limit the applicability of corpus-based methods to CLIR due to the scarcity of available parallel corpora. We shall see below that the divergence effect was quite pronounced when the retrieval collection differed from target-language section of the parallel corpus in one experiment reported in the literature.

We examined four different corpus-based query translation methodologies:

- Building a Lexicon from a Corpus
- Using Translation Probabilities with HMM-based Retrieval
- Similarity Thesauri
- Latent Semantic Indexing

(Some of these methods assume independence between individual terms in the source-language query. This assumption, although not strictly correct, is one commonly made in IR).

2.10.1 Building a Lexicon from a Corpus

This denotes building a lexicon or bilingual dictionary from a sentence-aligned (as opposed to document-aligned) bilingual parallel corpus. During the training phase, statistical methods are applied to match terms in source-language sentences with equivalents in their aligned target-language counterparts. These mappings form entries in the lexicon or part of the language model which is then used to translate queries. IBM pioneered this type of work for traditional MT with their statistical machine translation software [19] (see section 2.6). Here, we look at less sophisticated methods, seeking merely to map single concepts rather than to produce a full MT-style translation.

Hiemstra implemented this approach using a domain specific corpus but did not furnish any clear results [43]. Nie tested a very similar method, recording AvP of 25% for running some English queries

on a document collection consisting of French and English documents, and 27% for French queries on the same collection [68]. No corresponding monolingual results were provided. Davis found that a dictionary-based approach was superior to lexicon building alone, and recommended that the two approaches be combined for maximum effectiveness [27].

2.10.2 Building Translation Probabilities with HMM-Based Retrieval

Some researchers have incorporated translation probabilities derived from parallel texts in a similar manner to the bilingual lexicons described above into retrieval engines based on Hidden Markov Models (HMMs). The Haircut system at Johns Hopkins University employed translation probabilities derived from a Chinese-English parallel corpus at TREC-9, achieving from 25% to 49% of the corresponding monolingual performance [64]. Xu and Weischedel at BBN Technologies implemented a similar method, also at TREC-9, demonstrating that performance for this method was improved when a number of different resources were employed to derive the translation probabilities [103]. They recorded average precision of around 30%, but no corresponding monolingual performance value was supplied.

2.10.3 Similarity Thesauri

The Eurospider system of Similarity Thesauri was developed at ETH-Zurich by Ballerini *et al* [91]. This hinges on the inversion of the term-document matrix created by the Vector Space model of information retrieval when indexing a parallel document collection aligned at the document level [87]. Each pair of aligned documents is concatenated to form a single document. Then, instead of viewing the terms as indexing the documents, the documents are seen as indexing the terms. Terms that appear in similar documents are considered to be similar in meaning. This document-term matrix using the combined aligned document pairs is called a *Similarity Thesaurus*. The advantage of this method is that the documents only need to be aligned at the document, as opposed to the sentence, level, thereby reducing the cost in terms of both time and money of producing the aligned bilingual corpus.

Translation consists of query expansion. For each source-language query term, similar terms (terms which appear in aligned document pairs) are added to the query translation. Suitable target-language equivalents are obtained in this manner.

Ballerini *et al* applied this method to the same aligned subset of the SDA collection as was used in the LSI experiments described in section 2.10.4. Results of 20% AvP were recorded in TREC-7 [14], and 11% in TREC-8 [12]. Comparable monolingual retrieval results were not available.

The method was also applied by Ballerini *et al* to a pair of less similar collections - the German SDA collection and the English AP newswire [12]. Performance was considerably worse, probably due to the greater divergence between the two collections. The AP collection is described as not being "suitable" - its content was too different from that of the SDA collection. This demonstrated that unless the retrieval collection could be document aligned, or a document-aligned parallel collection very similar in content to the SDA collection could be found, this method does not perform very well. As document-aligned collections are thin on the ground, and hand-aligning a large document collection for each subject domain covered by the retrieval collection would be time-consuming and expensive (although less so than aligning the same collection at the sentence level), the applicability to CLIR of the Similarity Thesaurus method is limited.

2.10.4 Latent Semantic Indexing

Latent Semantic Indexing (LSI) was first proposed for CLIR by Young in 1994 [104]. LSI employs concepts borrowed from the field of linear algebra to reduce the dimensions of the term-document matrix constructed in the Vector-Space model of information retrieval [87] without loss of information, using a mechanism known as Singular Value Decomposition (SVD). SVD maps words occurring in similar contexts to points close to one another in the reduced dimensional space created [29]. This technology effectively defines a term using its surrounding context. In the case of LSI for IR, this context is the set of documents in which a given term occurs.

Young realised that this technology could be applied to CLIR. Documents in a bilingual parallel collection are aligned and a term-document matrix is built for both parts of this collection. The matrices

are combined (left-to-right) to form one and SVD is applied to this new matrix. Terms which appear in a similar context (in similar documents) are "folded" into the same part of the SVD space. Query terms are translated by selecting the target-language equivalent which is closest to the source-language term in this space.

Rehder *et al* created a document-level aligned subset of the SDA collections in a similar manner to IBM and used this aligned subset of documents to perform LSI for multiple languages (see section 2.6) [80]. Unfortunately the results using this quasi-parallel collection were not promising. LSI for CLIR was also tested by Mori *et al* for Japanese-English retrieval, with disappointing results [65]. They used a subset of the NTCIR retrieval collection as a parallel training corpus for the LSI algorithm and subsequently performed retrieval experiments using the entire NTCIR collection.

Because of its sensitivity to context, LSI needs a truly parallel (sentence-aligned) document collection to function effectively for CLIR. As truly parallel document collections are few and far between, this means that it is rarely practical to use LSI for CLIR.

We note here that Carbonnell *et al* implemented a method with many similarities to LSI called the Generalised Vector Space Model (GVSM) [20], with performance superior to that of LSI for CLIR. However, it suffered from exactly the same problems.

2.10.5 Concluding Remarks on Corpus-Based Methods

Although corpus-based approaches to query translation do not rely on hand-crafted knowledge to the same extent as MT-based methods, a significant investment is still necessary to produce a sentence- or document-aligned parallel corpus. In addition, these methods are sensitive to changes in subject domain - it is not clear that a corpus in a given language pair in one subject domain could be used to generate information to translate queries in another domain. This means that multiple parallel corpora could be needed for a single language pair. Since there is no guarantee that the resources needed to build a parallel corpus from scratch will always be available, we need to consider a family of approaches to CLIR which do not rely on hard-to-find specialist hand-crafted resources - dictionary based query translation.

2.11 Dictionary-Based Query Translation

Approaches to request translation which rely on commercial MT software and techniques which employ a parallel corpus to translate bag of words query terms both rely on resources which are not necessarily always available for the desired language pair and subject domain. The advantage of using a simple bilingual dictionary to translate query terms is that, although dictionaries themselves do have to be hand-crafted, usually by a team of lexicographers at the relevant publishing house, dictionaries and wordlists covering a wide range of subject areas and language pairs are readily available. In addition, the time needed to implement and set up a dictionary-based system from a printed or electronic source, although not null, is considerably less than for, say, extending an MT engine to a new language pair [47]. Therefore, our investigations have concentrated on a dictionary-based approach.

A *machine-readable bilingual dictionary* is defined as a data structure which contains a list of dictionary *entries* for a given set of terms, and a *lookup mechanism* which, given a source-language query term, consults this data structure to obtain a *bag* of one or more possible translations or *equivalents* of the term in question. An *entry* in a machine-readable bilingual dictionary is a data structure within a the dictionary containing all of the necessary information for a given spelling of a source-language query term. This information must include one or more *equivalents*. These equivalents may or may not be organised into sub-entries internally, for example, according to sense. In addition, other information, such as the part-of-speech of each equivalent, usage information, or translations of phrases containing the terms may be provided. We have restricted our definition of machine-readable bilingual dictionary to dictionaries which do not concern themselves with grammar or word order, and which do not interrogate the surrounding context of a term before providing a list of possible translations. This means that we do not include the complex lexical structures present in transfer-based MT engines in our definition of a dictionary.

Some terms, such as, for example, homographs like *lead*, as in *dog lead* and *lead pipe*, would have more than one entry in a conventional printed dictionary. In a machine-readable CLIR dictionary, multiple entries for a single spelling of a term tend to be conflated into a single entry as it is not always possible to

distinguish between homographs at run-time. Machine-readable dictionaries are typically derived from some other source, such as a printed edition of a published dictionary or an on-line wordlist. (Our own machine-readable dictionaries were derived from conventional printed dictionaries and are described in section 3.2). Various strategies are discussed in the literature for selecting a subset or *sub-bag* of equivalents for inclusion in the query translation from the bag of equivalents obtained for a given source-language query term, and for applying additional query term weights or *S-Weights* to those selected.

References to *R-Weights* in the rest of this document are to the weights calculated for each equivalent in the query translation by the retrieval engine term weighting mechanism at run-time, whereas *S-Weight* is defined as any additional explicit query term weight or equivalent weight applied at translation time. An S-Weight acts as a multiplier of the R-Weight. We subdivides S-Weights further into *Q-Weights*, which are S-Weights applied to query terms prior to query translation, and *T-Weights*, S-Weights applied to translation equivalents after dictionary lookup. Nearly all of the methods discussed in the literature concern T-Weights. Therefore, references to S-weights in the remainder of this chapter refer to T-Weights only unless otherwise stated.

Furthermore, we stated above that the lookup procedure will return a *bag*, not a *set* of equivalents for each source-language query term, meaning multiple occurrences of the same equivalent may be returned for a given term. Because of they way in which our retrieval system handles queries, allowing two or more occurrences of an equivalent to be added to the query translation is effectively the same as applying an S-Weight to that equivalent equal to the number times it occurs in the query translation. (A study of the effect of multiple occurrences of the same equivalent in a single CLIR dictionary entry was an important part of our research - see chapters 4 and 5).

We have divided the process of dictionary-based query translation into 4 logical stages:

- *Pre-Translation Query Modification.* This denotes any addition or deletion of source-language query terms or the application of any Q-Weight to any query term prior to translation.

- *Dictionary Lookup.* This is where the lookup mechanism of the dictionary is invoked to obtain a bag of equivalents for each source-language query term. As this "bag" is often considered to have an ordering imposed on it, we refer to it from now on as a *list* of equivalents. No selection or T-weighting of the equivalents is carried out at this stage. Any previously assigned Q-weights are transferred here.

- *Equivalent Selection and T-Weighting.* For each source-language query term, this is the selection of some or all of the equivalents obtained during dictionary lookup for that term for inclusion in the query translation, and/or any calculation of additional T-Weights. Allowing multiple occurrences of an equivalent to remain in the query translation constitutes a type of implicit T-Weighting for our retrieval system.

- *Post-Translation Query Modification.* This includes all explicit addition and deletion of equivalents in the query translation carried out after all explicitly translation-related operations have been completed.

Most research to date has focused on the equivalent selection and
T-Weighting stage, as the aim is to reduce the number of equivalents in the final query translation as much as possible without removing any "correct" translations. In particular, selection methods which involve large-scale processing of the retrieval collection or another resource have met with considerable success. We propose that there is another way to obtain a good level of performance for dictionary-based query translation without resorting to large-scale processing of the retrieval collection or of another resource, namely focusing on reducing the number of equivalents by choosing a dictionary with optimal characteristics, and reducing the number of terms which are translated in the first place. (It may not always be possible to implement large-scale processing of the collection in the case where the collection is both very large and dynamic, changing on a daily or hourly basis).

Our contention is that by proving a better "base" for equivalent selection by working on the first two stages, a similar level of performance to that reported in the literature for the more sophisticated selection methods can be obtained using very simple selection methodologies. We now present current research in each of the four stages, indicating those areas we investigated in our experiments and explaining why we chose to study them.

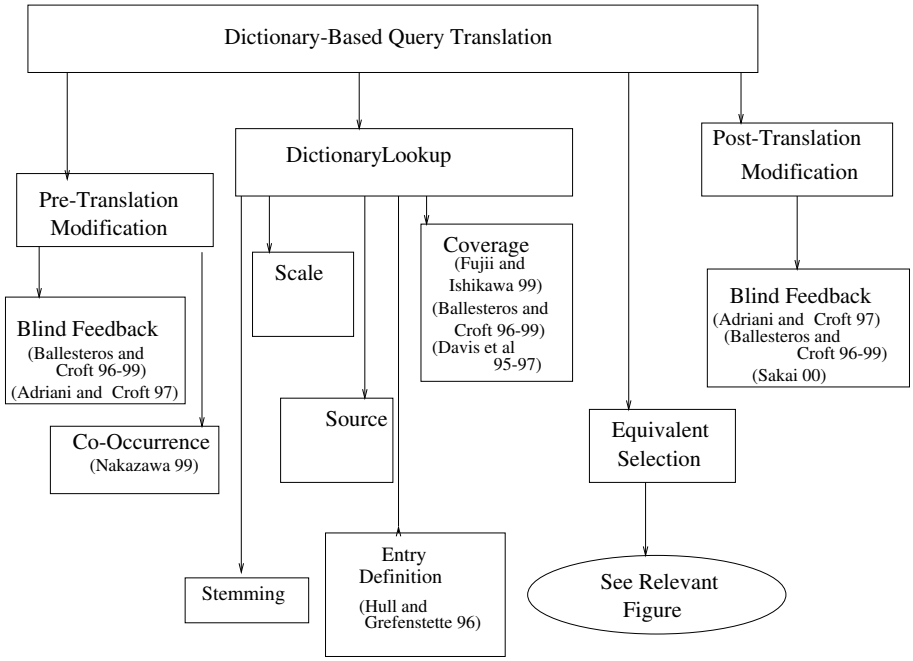

Figure 2.7: Stages of Dictionary-Based CLIR

2.12 Pre-Translation Query Modification

This stage comprises any explicit Q-Weighting, addition or deletion of any source-language query *term* before any translation is carried out, whether manual, automatic or a mixture of both. The assumption made here is that although a query may contain many terms, not all of them will be equally useful for retrieval. Terms can be deleted, for example, if it is considered that the inclusion of any translation of a given term in the query translation would harm retrieval. Alternatively, adding some new terms to the source-language query prior to translation might also help retrieval performance after translation. Finally, Q-Weights reflecting the importance of a query term in determining the query content could be applied to the terms in the source-language query before translation. In this case, the Q-Weight applied to a given term would also be applied to any equivalent of that term included in the query translation after translation has been carried out. Where a T-Weight is applied to one of these equivalents at translation time, the product of both the term Q-Weight and this new T-Weight comprises the final S-Weight applied at retrieval time to that equivalent.

The most popular automatic modification method has been the application of a *blind relevance feedback* step using a document collection in the source language. (For a description of how blind relevance feedback works, see section 2.9). This strategy has been quite successful [3, 9].

Adriani and Croft hand-translated the TREC-5 Spanish topics to Indonesian to perform English-Indonesian CLIR [3]. Prior to translation, a blind relevance feedback step was applied to both the English and Indonesian versions of the queries. The collections used to perform the feedback step were distinct from the retrieval collections.

It was discovered that for English-Indonesian retrieval, the feedback step was not particularly helpful, as there were not a sufficient number of documents pertaining to Indonesian affairs in the English-language collection used to perform the feedback step - in fact, performance dropped when pre-translation feedback was implemented. On the other hand, there were plenty of documents in the Indonesian collection which dealt with international affairs and an improvement in performance of 10-15% was recorded for Indonesian to English retrieval after the implementation of blind relevance feedback.

This indicated that pre-translation feedback should be performed only when a pertinent source-language document collection is available. This is because when there are relatively few, or no, documents in the source-language collection used for feedback which are relevant to a given query, not many of the top N documents retrieved by the feedback step will actually be relevant or even in the correct subject

area. Consequently many irrelevant terms may be added to the source-language query (providing the relevance feedback method employed allows new terms to be added) leading to an explosion of "garbage" equivalents in the query translation, requiring stringent equivalent selection methods to be applied later on. Where there are many relevant documents in the source-language collection, this is less of a problem, as more of the terms added during feedback are likely to be from relevant documents or documents in a related subject area and therefore more likely to be pertinent to the query.

Ballesteros and Croft reported similar findings [7]. (They called blind relevance feedback *local feedback*). Ballesteros and Croft also experimented with a technique they called *Local Context Analysis* [102]. This was similar to adding new terms to the query using blind relevance feedback except that in considering which terms from the top N retrieved documents to add, only those document terms which appear in the same *context* as a given source-language query term in these top N documents were considered. By appearing in the same context, they meant occurring within M terms of the relevant source-language query term, where M was an arbitrary value set by the experimenter. Performance when Local Context Analysis was applied was found to be superior to that of blind relevance feedback in certain circumstances [9].

Nakazawa *et al* tested an alternative to blind feedback, implementing query expansion using *synonyms* [66]. Synonyms were defined as terms which appeared frequently close together in documents in a source language document collection which was comparable to the retrieval collection. The concept of exploiting *co-occurrence information* or *collocation information* has been a widespread practice in CLIR in recent years, where query expansion terms or translation equivalents are selected based on how frequently they appear within X terms of each other in the document collection. Nakazawa's synonym selection method is described here in detail to provide an illustration of how this type of analysis is carried out. The main area in which co-occurrence or collocation data has been employed in CLIR is in selecting translation equivalents.

When one says a term *co-occurs* with a given source-language query term or appears with it within a *collocation* in the source-language collection, that means that it appears close to it more frequently in the collection than would be due to chance alone. Closeness of a term A to a term B can be defined as appearing within N terms of one another, or as appearing in the same document. In the case of Nakazawa's method, and of most of the co-occurrence/collocation information exploitation methods discussed in the literature, we use the metric "within N terms" where N is arbitrarily chosen by the experimenter. Nakazawa calculated the *sum of co-occurrence or collocation frequencies* of a source-language collection term j_a with all source-language query terms $j_1 \dots j_n$, where there are n source language query terms, using the formula:

$$\sum_{i=1}^{n} \frac{f(j_i, j_a)}{f(j_i) \cdot f(j_a)}$$

where $f(j_x)$ is the number of times term x occurs in the collection, and $f(j_x, j_y)$ is the number of times terms x and y occur with N terms of one another in the collection. Numerous other formulae exist and are discussed in section 2.12.

Terms whose sum of co-occurrence or collocation frequencies exceeded a certain threshold were added to the source-language query before translation. However retrieval performance (AvP) for all of Nakazawa's test runs reported was so low that it was difficult to pinpoint the effects of this technique on retrieval performance.

The problem with co-occurrence/collocation methods for term Q-Weighting is that they depend on an additional resource - a closely-matching source-language document collection. Whereas such a resource is easier to obtain or create than, say, an aligned parallel corpus, it nevertheless represents additional effort which it might be possible to avoid by employing query modification techniques which do not need a similar source-language collection to function.

In our own experiments, We investigated the deletion and the Q-weighting of query terms based on their degree of ambiguity in the dictionary. We defined the *degree of ambiguity* of a term as being the number of distinct target-language equivalents listed in its entry in a designated bilingual dictionary. (The choice of designated dictionary will obviously have an impact on results). The aim of this work is to see how much leverage we can exert on retrieval performance of query translations without consulting any resource other than the dictionary itself.

2.13 Dictionary Lookup

Dictionary lookup is the second stage of dictionary-based CLIR and the first step of the actual translation, or concept mapping, process. For each source-language query term, we obtain a list of target-language equivalents from the dictionary consisting of all the equivalents listed in that term's dictionary entry. No attempt is made to select or T-Weight any of the equivalents at this stage. (Although a given list of equivalents may contain multiple occurrences of the same equivalent. The implicit T-Weighting inherent in such duplication is discussed in section 2.14.2).

Some translation methodologies, such as taking the first equivalent listed in the dictionary for each term, appear to combine this stage with equivalent selection. However, we may still view this as two distinct logical steps - obtaining the entire equivalent list from the dictionary for each term (dictionary lookup), and then selecting the first equivalent from each term's list for inclusion in the query translation (equivalent selection).

Researchers in dictionary-based CLIR have employed a variety of dictionaries for lookup purposes. Some have been obtained from standard printed dictionaries, both unabridged and pocket-sized, others from free wordlists downloaded from the Internet [8, 67, 57]. We consider the characteristics of the dictionary to have an important influence on the retrieval performance of query translations. We maintain that a good equivalent selection method does not compensate for a poor choice of dictionary or dictionaries. The effect on retrieval performance of a number of dictionary characteristics needs to be assessed:

- *Coverage*: The overall coverage rate of the dictionary.
- *Entry Definition*: How a dictionary entry is defined.
- *Stemming and Lemmatisation*: The potential impact of stemming or lemmatisation of dictionary entries.
- *Dictionary Scale*: The average number of distinct equivalents in each dictionary entry.
- *Minor Variations in Content*: The effect on retrieval performance of small differences in query translation composition between translations obtained from very similar dictionaries.
- *Equivalent Repetition Within Dictionary Entries*: The effect on retrieval of allowing multiple occurrences of a given equivalent in a single dictionary entry.

A summary of current research in each of these areas, and the avenues we chose to explore, is presented below.

2.13.1 Coverage

The *coverage rate* is the percentage of terms in the query set which are listed in the dictionary. If a given term is not listed in the dictionary, a translation for it will not be found. A high rate of coverage is very important for retrieval performance [47]. Ensuring a rate of coverage as close to 100% as possible, especially for technical usage, has been the focus of considerable research effort. Davis found augmenting a dictionary with translation correspondences gleaned from a parallel corpus to be beneficial for retrieval [27]. Ballesteros and Croft exploited the phrase and usage information in a large printed dictionary to find translations for multi-word terms in source-language queries [9]. Fujii and Ishikawa employed a technical term dictionary on top of a general Japanese-English dictionary in NTCIR1 [34]. All of these experiments met with some success, therefore, we concluded that further research to corroborate them was not necessary. In our experiments, we ensured that any dictionaries used had a coverage rate of 100%, to prevent lack of coverage introducing artifacts into experiments designed to study the effect of other factors in isolation.

2.13.2 Entry Definition

Performance may also be affected by the manner in which CLIR dictionary entries is defined or derived. For example, the process whereby Hull and Grefenstette derived a CLIR dictionary from an electronic copy of a printed, university-level English-French dictionary resulted in the presence of "garbage" equivalents in entries which were not in fact translations of the corresponding term at all, resulting from errors in

the electronic source [47]. For example, there were cases of words from the usage example sections of the electronic source entries being included as equivalents in some cases. It was thought that the presence of these "garbage" equivalents could have had an unmeasured negative effect on their results. However, most experimenters have not supplied this level of detail in their reports, making it difficult to compare results. In our experiments, we created test dictionaries by hand, containing entries for the 385 terms in our test query set only, thus ensuring that no "garbage" equivalents were present. A working system would need to address the problem of ensuring the CLIR dictionary entries were "clean". We lacked the resources to compare a wide-coverage "clean" dictionary with one which was identical but contained some "garbage" equivalents. This would be an interesting experiment for those in possession of such a dictionary to carry out, although it would require extensive hand-editing of thousands of entries.

2.13.3 Stemming and Lemmatisation

As already mentioned above, stemming is the removal of suffixes from terms in queries and documents, whereas lemmatisation is the conversion of terms to their canonical dictionary form. In order to allow source-language query terms to be looked up in the dictionary in our system, source-language queries must be lemmatised. However, it does not follow automatically that we must also lemmatise the target language collection and the target language equivalents contained within our dictionary or dictionaries.

It is standard practice in information retrieval to normalise all terms in the queries and the retrieval collection. By normalisation, we mean stemming, lemmatisation, and operations such as decompounding in German. Stemming is one of the most widely normalisation methods - a description of a popular stemming algorithm may be found in Porter 1980 [76]. No study to date has compared the effect on retrieval performance of stemming the retrieval collection versus lemmatising it. We chose to lemmatise both the target language collection and the target language contents of our dictionary entries as a matter of personal taste. When we began our experiments, it was not clear whether or not we would also have to build a French-English dictionary by hand in a similar manner to our English-French dictionaries (see chapter 3 for details on the latter). A stemmed equivalent is likely to match several dictionary entries, making the task of building a French-English dictionary more onerous and time consuming than if we were dealing with lemmatised equivalents. As it happened, we did not construct a French-English dictionary, but it was not clear at the outset that this would not be necessary. Hence, we chose to lemmatise all target language content in our English-French dictionaries and also lemmatised the French retrieval collection.

2.13.4 Dictionary Scale

Dictionary scale is a measure of the average number of equivalents and senses in a dictionary entry. Although the different dictionaries mentioned in the literature are unlikely to all have been of the same scale, no formal measure of scale has been defined and no direct comparative experiments regarding the effects of varying dictionary scale on the retrieval performance of associated query translations have been carried out. We consider the issue of dictionary scale to be very important as this more than anything else influences the number of equivalents returned for each term on dictionary lookup. A dictionary of smaller scale will return fewer equivalents, but is more likely to omit an important equivalent. The combined effect of these two factors on retrieval performance needs to be assessed. Therefore, the issue of dictionary scale was the first to be tackled.

2.13.5 Minor Variations in Content

The dictionary employed by a CLIR system can be obtained from a variety of different sources - for example, a printed tourist dictionary, a spell checking algorithm or other piece of software, an on-line reference dictionary ... We wished to find out if small differences in equivalent lists reflecting the idiosyncrasies of individual lexicographers' decisions for dictionaries of similar scale derived from similar sources had a significant effect on retrieval performance. This is important as if these small differences do affect performance, great care must be taken when choosing a dictionary, even when the future CLIR dictionary's scale and type of source have already been decided upon.

2.13.6 Equivalent Repetition Within Dictionary Entries

Printed dictionaries aimed at language learners, especially those which aim to be more complete and in-depth, tend to list the same possible translation of a given term more than once within its printed entry, because it is the best translation of more than one sense of the source-language term. When we derive a CLIR dictionary from such a source, we can retain such repetition within the CLIR dictionary's entries' equivalent lists, or we can discard any such repetition. Given that our retrieval engine considers queries submitted to it to be a flat, unstructured list of independent terms (see chapter 3), allowing repetition to remain in a query translation amounts to implicitly applying an additional T-weight to the repeated equivalents. We wanted to know what kind of implicit T-weighting was good for retrieval performance, and determine the cases where allowing repetition to remain in the query translation harmed retrieval performance. This would provide a policy for dealing with CLIR dictionaries derived from sources whose entries contained multiple occurrences of equivalents. It must be emphasised that this multiplication of the T-weight is dependent on our query processing mechanism, however, this is a common method of processing queries and many other engines also deal with queries as flat term lists unless special operators are invoked. For a system which dealt with more structured queries only, we would need to adapt our experiments.

Our work on equivalent repetition led us to consider additional simple S-weighting strategies (both Q- and T-Weighting). Current research on equivalent selection and T-weighting is presented below, along with a description of the types of T-weighting we investigated and why. We note that we did not tackle the possibility and effect of equivalent repetition within a query translation due to being provided as a translation for more than one term. As this type of repetition is less common and harder to predict, we chose to focus on repetition within dictionary entries only. Therefore, queries "without repetition" are so called because they are obtained using dictionaries containing no repetition in their entries - it is not guaranteed that each equivalent in the resulting query translation occurs only once within the entire translation, just within the list of equivalents provided by the dictionary for a given term.

2.14 Equivalent Selection and S-Weighting (T-Weighting)

It is desirable to reduce the number of equivalents in the query translation as much as possible, without deleting those which are crucial to the retrieval performance of the translated query [47]. Numerous strategies for selecting a subset of equivalents from the lists provided by the dictionary lookup mechanism for each source-language query term have been proposed, along with techniques for calculating T-Weights for those that have been selected. We have divided selection and T-Weighting strategies into four categories:

- *Selection and/or T-Weighting Based Solely on Dictionary Information.* This is where no information other than that contained in the dictionary itself is employed to perform selection.
- *Hybrid Weighting Using Equivalent Grouping.* This is a hybrid of S- and R-Weighting which can be applied in conjunction with any selection method.
- *Selection and/or S-Weighting Based on Information from the Retrieval Collection.* This is where the target-language retrieval collection as well as the dictionary can be exploited to select and S-Weight equivalents.
- *Selection and/or T-Weighting Using Another Resource.* This is where a resource other than the dictionary and the target-language retrieval collection, such as a parallel corpus, is employed to select and/or T-Weight equivalents.

2.14.1 Selection and/or S-Weighting (T-Weighting) Based Solely on Dictionary Information

In this section, we shall describe equivalent selection and T-Weighting techniques which do not consult any resource other than the dictionary itself. These are the simple selection methods mentioned above. We have divided these methods into the following categories:

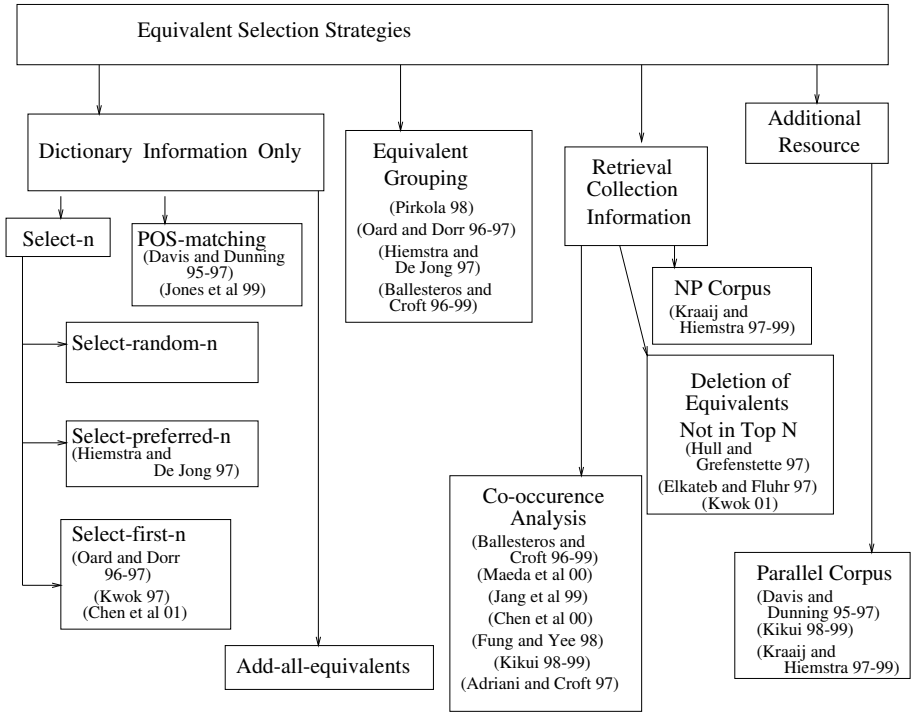

Figure 2.8: Equivalent Selection Strategies

- 2.14.2 *Add-All-Equivalents*. This is where no selection is performed, and T-Weighting may or may not be implemented.
- 2.14.3 *Select-N*. This is where a sub-bag of n equivalents is selected from each term's list of equivalents for inclusion in the query translation.
- 2.14.4 *POS (Part-Of-Speech) Matching*. This is where the part of speech of the source-language query term is used to constrain the set of equivalents added to the query translation.
- 2.14.5 *Ambiguity-Based Selection and T-Weighting*. This is where equivalents are retained, deleted or have a T-Weight assigned to them in the query translation based on their *degree of ambiguity*.

The next four subsections present these methods in detail.

2.14.2 Add-All-Equivalents

This is the null selection method. The lists of equivalents obtained during dictionary lookup are added to the query translation without modification. This method does not perform very well. Its main purpose in experiments is as a baseline with which more sophisticated selection methods can be compared [72]. It may also be accompanied by some form of T-Weighting, explicit or otherwise.

Our definition of an equivalent list and a dictionary entry allows multiple occurrences of a given equivalent within a single entry or list. Using our retrieval engine, adding multiple occurrences of a given equivalent to the query translation is effectively equivalent to assigning the repeated equivalent an implicit doubled T-Weight. In our own experiments, we define Add-All-Equivalents as the null method where no selection is performed, no special operators are invoked to break the query up into groups of terms and *no T-Weights are applied*. Therefore, any subsequent references to Add-All-Equivalents assume that no S-Weights (including, in the case of our own retrieval engine, repeated equivalents) have been applied unless otherwise stated.

For TREC-7, Hiemstra and Kraaij applied T-Weighting to equivalents according to how many times an equivalent appeared the dictionary entry of the source-language query term under consideration [45], while including all distinct equivalents in the query translation. Variations of strategy combined with fuzzy

expansion of query terms achieved between 73% and 92% of the corresponding monolingual performance. These results are comparable to those obtained for the more complex selection methods discussed below, but as the strategy was combined with fuzzy expansion it is difficult to determine whether the selection method used was actually solely responsible for these results.

2.14.3 Select-N

The objective of this group of strategies is to reduce the number of equivalents in the query translation by selecting n equivalents from the list provided by the dictionary for each source-language query term. These n equivalents are selected according to some criterion. These methods may be combined with T-Weighting techniques which target those equivalents which have been selected.

Select-First-N involves taking the first n equivalents from the *front* of the list returned by the dictionary. We stated above that we use the expression *equivalent list* instead of *bag of equivalents* because the dictionary will have imposed an ordering on the equivalents by virtue of where they occur in the corresponding entry. This selection method assumes that the most useful and common equivalents will be found at the front of the list, an assumption which is usually correct when the dictionary entries have been derived from a standard printed bilingual dictionary, the most common source of CLIR dictionaries cited in the literature.

Oard found that setting $n = 1$ resulted in the best retrieval performance of a number of CLIR methods [72]. Kwok achieved the highest level of performance in a different experiment by setting $n = 3$ [56]. Chen *et al* implemented a method which implemented Select-First-N with $n = 2$ [22], but as it was combined with phrase recognition techniques its impact cannot be determined from the results provided. No experiments providing detailed results for different values of n using the same data and retrieval engine have been reported.

Select-Random-N is where the n equivalents are selected at random from the list of equivalents returned by the dictionary lookup mechanism for each source-language query term. Care must be taken when randomly selected data is being employed to ensure that this randomisation process is not introducing artifacts in the results. The main purpose of this method is to demonstrate that the assumption of an ordering being imposed by the dictionary made for Select-First-N above is valid. This would serve as an explanation for why Select-First-N appears to perform so well for lower values of n. No experiments comparing Select-Random-N with Select-First-N for various values of n have been reported.

Select-Preferred-N also involves taking the first n equivalents from the front of each source-language query term's equivalent list, but after a different ordering function has been applied. The equivalents obtained during lookup for a given source-language query term are ordered in descending order of how many times they occur within the term's dictionary entry. This method is based on the assumption that the equivalents which occur the most frequently are those which are the most commonly employed and will constitute the best translation. If this assumption were true, results for Select-Preferred-N would surpass those for Select-First-N or Select-Random-N. Hiemstra and De Jong implemented this method, obtaining 81% and 72% of the corresponding monolingual AvP for selecting the top n preferred translations, with $n = 1$ [44].

No experiments involving selection for values of n greater than 1 have been reported, nor any comparing the effectiveness of the different Select-N strategies.

2.14.4 POS-Matching

This method assumes that the dictionary lookup procedure also provides information on the *Part Of Speech* (POS) of each equivalent, for example, whether it is a verb or a noun. It involves retaining only those equivalents whose part of speech match that of the source-language query term for inclusion in the query translation. Davis found that POS-matching improved retrieval performance for English-Spanish retrieval by 36% [28]. Jones *et al* found POS-matching resulted in performance similar to that recorded for select first n, $n = 1$ for Japanese-English retrieval [50]. Our own dictionaries did not record POS information and so we did not perform any POS-matching experiments.

2.14.5 Ambiguity-Based Selection and T-Weighting

This is where information contained in the dictionary regarding the degree of ambiguity of a given equivalent is used to decide whether or not to add the equivalent to the query translation and whether or not a T-Weight needs to be applied. The degree of ambiguity of an *equivalent* (as opposed to a source-language query term) is defined as the number of distinct translations listed for it in a dictionary for translation from the target language to the source language. This measure depends on the dictionary used to calculate it. The strategy is applied after dictionary lookup, and is distinct from any ambiguity-based deletion or Q-Weighting carried out on the source-language query prior to translation. Although strictly speaking this method does consult a resource other than the source to target language bilingual dictionary, it is reasonable to assume that any dictionary obtained which will translate from the source to the target language is also able to translate from the target language back to the source. In the absence of such a dictionary, the existing dictionary can be inverted. (Hiemstra and Kraaij inverted the Van Dale Dutch dictionaries for their CLIR experiments [45]). No results have been reported in the literature concerning this selection technique. Therefore, we performed some experiments using simple ambiguity-based equivalent T-weighting methods.

We now move on to the next type of equivalent selection technique, hybrid weighting using equivalent grouping.

2.14.6 Hybrid Weighting Using Equivalent Grouping

This is a weighting-only method which may be combined with any of the selection methods described above or implemented on its own. Instead of applying additional T-Weights to the equivalents in the query translation, the R-Weight calculated by the retrieval engine is constrained by the application of a maximisation function. The query translation is divided into *groups* of equivalents according to the source-language query term of which an equivalent is a possible translation. These groups are then each treated as a single equivalent for retrieval purposes, with each occurrence of any equivalent in a given group in a document at retrieval-time being viewed as an occurrence of that group. The R-Weight assigned to a given group is the maximum R-Weight calculated for any equivalent in the group by the retrieval engine. This is similar to applying a query-term weight to an OR clause in the extended Boolean retrieval model [86]. The aim of equivalent grouping is to avoid the situation where documents matching on many equivalents of a single source-language query term are ranked above more relevant documents in the retrieved document list.

Pirkola's technique [75] used the *facet* operator of the INQUERY retrieval engine [18] to achieve equivalent grouping, recording a performance improvement of 50%. Oard also applied this technique in TREC-8 [72]. Ballesteros and Croft used the INQUERY synonym operator to perform the same operation [10]. Results assessing the effect of this operation on retrieval performance in isolation were not provided.

Hiemstra and De Jong implemented a similar method, which they called *query structuring* [44]. They tested this method using Add-All-Equivalents (non-)selection without explicit or implicit T-Weighting, and also combined grouping with other T-Weighting and selection methods. Varied results were obtained. For example, 31% AvP was obtained for grouping combined with Select-Preferred-N compared to 34% for combining grouping with equivalent selection using a parallel corpus (see below). Grouped translations performed better than their ungrouped counterparts in all cases.

It follows that implementing equivalent grouping in a dictionary-based query translation system would be desirable, but it should be applied in conjunction with the best available selection method, as the choice of selection method also affects retrieval performance. We have not implemented equivalent grouping in our experiments, as we wanted to observe the effects of various factors on retrieval performance in isolation. However, if we were to construct a working system, we would certainly include one of these equivalent grouping techniques.

2.14.7 Selection and/or T-Weighting Based on Information from the Retrieval Collection

This section describes equivalent selection and T-Weighting methods which consult the dictionary and the target-language retrieval collection only. It does not include approaches to query translation which also use a comparable retrieval collection or parallel corpus, which are dealt with in section 2.10.

The main approaches in this area are:

- 2.14.8 *Calculation of Co-Occurrence Frequencies.* This is where a formula for calculating the *sum of co-occurrence or collocation frequencies* is used to rank the results of Add-All-Equivalents selection and the top N equivalents in this ranked list included in the query translation.
- 2.14.9 *Deletion of Equivalents not in Top N Documents.* This is where a pseudo-relevance feedback step is performed and any equivalents not present in the top N documents retrieved by the feedback step discarded from the Add-All-Equivalents query translation.
- 2.14.10 *Noun Phrase List Translation.* Information about the noun phrases contained in the retrieval collection is used to compile the query translation.

The next three subsections present these approaches in detail.

2.14.8 Calculation of Co-Occurrence Frequencies

In section 2.12, we defined *co-occurrence* as two terms appearing within the same document or within N terms of each other in a document. We stated that both the expressions *co-occurrence* and *collocation* information could be employed to describe this kind of information. In this section, for the sake of clarity, we will use the expression *co-occurrence information* to describe both kinds of information. We saw in section 2.12 a detailed example of how co-occurrence information may be gathered from a document collection for a given pair of terms, and showed how a co-occurrence measure could be employed to rank a list of source-language query terms. Here, we examine methods where similar metrics and the retrieval collection are employed to rank equivalents. The sum of co-occurrence frequencies of all equivalents in the Add-All-Equivalents query translation (without any S-weighting, implicit or otherwise) with one another in the retrieval collection is calculated and these equivalents ranked accordingly. The top N equivalents in this ranked list are then included in the new query translation. The assumption made by this method is that groups of equivalents which co-occur frequently are likely to be related in terms of their subject domain. The chance of there being a large number of equivalents in the Add-All-Equivalents query translation related to a subject domain which is not relevant to the source-language query is quite small, therefore, equivalents which co-occur frequently with many others are more likely to be related to the source-language query content and will therefore be nearer the top of the ranked list of equivalents. This is the single most successful selection method reported in the literature, and has been widely implemented.

A description of one possible formula which can be used to calculate co-occurrence frequency is given in section 2.12. Numerous alternative formulae have been reported in the literature [10, 4, 62]. Ballesteros and Croft [10] combined a co-occurrence method using a variation of the EMIM metric [97] to rank equivalents combined with equivalent grouping and POS-matching, obtaining average precision of 30%, compared to 24% for Select-First-N, $n = 1$, and 26% for selection using a parallel corpus.

Maeda conducted a similar experiment using the mutual information measure [24] to calculate co-occurrence frequency for CLIR using documents on the World Wide Web [62]. AvP from 11% to 16% using the NTCIR collections was recorded, which constitutes close to average performance when compared with other experiments employing the NTCIR collections, although the usual caveats apply. Chen, Lin and Lin [23] employed a mutual information measure to create contextual vectors before consulting a Chinese-English extension of Wordnet [31], with disappointing results. Fung and Yee used a variant of a Salton and Buckley's cosine-based similarity measure [35]. No concrete results in terms of retrieval performance were given. Kikui opted for a measure called *coherence* to calculate co-occurrence frequencies [51], claiming over 80% translation accuracy. Adriani and Croft tested a method which performs co-occurrence analysis implicitly using a similarity measure based on the standard weighting formulae implemented in their retrieval engine [4]. AvP of 82%, 61% and 71% percent of monolingual English retrieval were recorded for German, Spanish and Indonesian to English retrieval respectively. These results are similar to those quoted for some MT-based query translation experiments.

Although advances in computing technology make it easier to implement co-occurrence-based selection, the main drawback is that unless considerable computing resources are available, calculating co-occurrence frequencies for each pair of terms in the collection remains rather effortful and expensive. In addition, some collections are dynamic, changing on a daily or hourly basis, such as a newswire archive, for example, making large-scale processing of the entire collection every day a somewhat onerous and expensive task. In our work, we have attempted to discover if all this calculation and processing can be circumvented by applying simple selection methods in combination with any insights gained in our examination of dictionary characteristics and pre-translation query modification.

2.14.9 Deletion of Equivalents not in Top N Documents

This method is a type of relevance feedback implemented by Elkateb and Fluhr in their EMIR system [32, 11]. An initial retrieval run is performed using the Add-All-Equivalents translation of the query. Then, all equivalents not found in the top N documents returned by this run are removed from the query translation. It is assumed that the top N documents, if not relevant, will at least be in the right subject area. However, Add-All-Equivalents translations have a tendency to retrieve a great many wholly unrelated documents because of the volume of equivalents in the query, so this may not always be the case. Average precision of 69% of the corresponding monolingual retrieval was recorded at TREC-6 for this method. These results are not as good as those quoted for co-occurrence frequencies above, but a lesser volume of calculation is required.

Kwok *et al* implemented a slightly different variation on this theme - the deletion of all equivalents which occurred less frequently than N times in the retrieval collection [58]. The maximum number of equivalents retained for any one source-language query term was 6. Retrieval performance of 69% of monolingual retrieval was recorded, the same as that observed by Elkateb and Fluhr. Implementing these selection methods is relatively straightforward, a simple frequency count suffices. It remains to be seen whether our work results in better performance using simpler selection methods.

2.14.10 Noun Phrase List Translation

This method was developed by Kraaij and Hiemstra [54]. The retrieval collection is parsed and a list of its constituent noun phrases compiled. When translating a query, the system searched for the constituents of each noun phrase in this list in the query's Add-All-Equivalents translation, and included in the final query translation all noun phrases thus found. This method did not perform any better than the much simpler Select-Preferred-N, $n = 1$ method detailed in section 2.14.3, but requires considerably more effort and computational time to implement.

2.14.11 Selection and/or T-Weighting Using Another Resource

A small number of researchers used a parallel corpus to aid equivalent selection. Equivalents are selected from the lists provided by the dictionary by exploiting the implicit usage information present in the parallel corpus. These methods have had some success, but suffer from the same problems as approaches to CLIR which rely exclusively on a parallel corpus, namely, the scarcity of available corpora. In addition, the extra effort does not appear to be justified as performance superior to that recorded for co-occurrence frequency calculation has not been observed for these methods.

Davis used a parallel corpus to aid disambiguation in English-Spanish retrieval [28]. However it was not clear which aspects of the improved retrieval performance were due to the POS-matching selection technique employed and which were the result of using the parallel corpus. Kraaij and Hiemstra found a parallel corpus to be the best equivalent selection method [55]. Kikui used a parallel corpus for word sense disambiguation using distributional clustering with promising results [52].

The scarcity of results and the effectiveness of methods which use the retrieval collection and the dictionary only seem to suggest that there is no real need to add a parallel corpus in order to obtain a reasonable level of retrieval performance for dictionary-based query translation.

2.14.12 Concluding Remarks on Equivalent Selection and T-Weighting

In this section, we have demonstrated that the selection method of choice for dictionary-based query translation should be co-occurrence frequency calculation, coupled with equivalent grouping. Where this selection method cannot be implemented, some simpler yet quite effective methods involving equivalent deletion are available. There is no need to invoke a further resource, such as a parallel corpus, as this does not appear to improve on the performance of co-occurrence frequency calculation for equivalent selection. However, it is not always practical to carry out the large volume of calculation needed for a co-occurrence frequency based equivalent selection technique, and existing simple dictionary-only methods are not as effective as one might like them to be. Therefore, we have concentrated our attention on testing some dictionary-information-only ambiguity-based selection and S-weighting methods.

2.15 Post-Translation Query Translation Modification

It is important to distinguish between the selection and S-Weighting methods described in the previous section, and query modification carried out on the query translation as if it had been issued in the target language to begin with, dealt with here. The idea here is that once all translation has finished, standard query enhancement mechanisms from the realm of monolingual IR can also be of benefit to our translated queries.

The most popular post-translation modification method is blind relevance feedback (see section 2.12). Post-translation feedback was discussed in section 2.9 with respect to query MT. It has also been found to be helpful to retrieval after dictionary-based query translation.

Adriani and Croft discovered that although the performance improvement due to post-translation feedback was greatest when there were many relevant documents in the collection, it was also beneficial when this was not the case [3]. Hence the caveat associated with pre-translation feedback (see section 2.12) does not apply here. Ballesteros and Croft found that post-translation feedback led to considerable retrieval performance improvements [8]. A rise of 28-47% in AvP was recorded for queries that had been translated by hand. Local Context Analysis (see section 2.12) gave rise to similar increases in AvP.

Sakai *et al* pointed out that although a blind relevance feedback step can increase average performance, it can nevertheless harm retrieval performance for approximately one third of the queries to which it is applied [83]. Various methods for improving the reliability of blind relevance feedback, such as tailoring the parameters of the weighting formulae used to individual test requests, were investigated.

However, post-translation feedback does not nullify the need for good-quality procedures in the three previous stages of dictionary-based CLIR. Ballesteros and Croft found that the application of a post-translation feedback step does not raise performance to the same level for all runs - relative performance differences prior to feedback were maintained after its application, although absolute performance increased [8].

Therefore, it seems that one's research efforts would be better employed in improving the first three stages of dictionary-based CLIR. Hence, we have not done any work in this area.

2.16 Conclusions

In this chapter, we reviewed the principal CLIR methods reported in the literature, assessing them with respect to retrieval performance, reliance on hand-crafted resources and range of applicability. We saw that dictionary-based query translation constituted the method with the greatest range of applicability, was the least reliant on hand-crafted resources and one of the easiest to implement. Although it was not the CLIR method which performed the best, nevertheless, a satisfactory level of performance has been reported in the literature for this method in several cases.

Document translation was found to be cumbersome, requiring the translation of every document into every language represented in the system. This would be costly and effortful if the collection were very large or in any way dynamic. In addition, document translation does not offer a great deal in terms of performance benefits over request translation.

We then examined approaches to CLIR which transformed both documents and requests to a third representation which could be searched directly. We found that these methods did not offer any performance improvements over document translation and required a significant number of hand-crafted knowledge sources, for example, Wordnet, to function. In addition, extending these resources to cater for new language pairs or domains would be a costly and time-consuming exercise, thereby limiting the applicability of these approaches. Therefore, one must consider techniques for translating the request or the derived query into the language of the retrieval collection.

The single best performing CLIR method reported in the literature is the translation of requests using commercial off-the-shelf MT software. In particular, systems which combined several MT engines or added a post-translation relevance feedback step recorded performance up to 97% of the corresponding monolingual run. However, MT technology is inherently limited by its dependence on its large, complex and hand-crafted internal lexica and resources. MT engines are currently available for just a few language pairs, and extending current systems to new language pairs requires the dedication of several man-years to the task. This means that we cannot always implement request MT.

One alternative is approaches to CLIR which borrow ideas from the field of corpus linguistics to translate the terms in the bag of words query derived from the user request. These corpus-based translation methods use a bilingual parallel corpus to extract translations for source-language query terms automatically. This means that the translation information does not need to be encoded by hand. However, the bilingual parallel corpora need to be aligned by hand and research into automatic alignment at the sentence level is on-going. This means that although the human effort needed to get a corpus-based system up and running for a new language pair would be significantly less than for any form of MT, it can still be prohibitively expensive if a large new corpus has to be obtained and aligned from scratch. Finally, there is some evidence that suggests that the bilingual parallel corpus needs to be very close to the retrieval collection in terms of content and subject matter for a corpus-based translation method to work effectively. This makes it even more likely that for a given language pair and subject domain, a new parallel corpus needs to be compiled and aligned.

This led us to consider dictionary-based translation of bag of words query terms. By dictionary-based, we mean a system which has at its core a simple bilingual dictionary similar to that a language learner might use. Since bilingual dictionaries are already available for most language pairs, it follows that the single hand-crafted resource such a system needs to function has already been compiled. Therefore, to implement a dictionary-based system one need only find a way of converting an electronic version of a traditional bilingual dictionary to a format that can be used to translate query terms directly. Although the work involved in converting a dictionary is not negligible, it is still considerably less than that required to extend any of the methods discussed above to a new language pair. In addition, many experiments using dictionary-based approaches have achieved a reasonable level of performance.

We concentrated our research in this area, due to the wide applicability of dictionary-based approaches. We divided the process of translating a query using a dictionary into four stages: pre-translation query modification, dictionary lookup, equivalent selection and T-Weighting, and post-translation query modification. We noted that effective methods for post-translation modification already existed, and so there was no need for us to do any more work in that area. Furthermore, the majority of research into dictionary-based CLIR has concentrated on the equivalent selection stage. This is because the way to get good retrieval performance using this method is to reduce the number of equivalents in the query translation as much as possible without eliminating the "correct" translations of the query terms. Many approaches to equivalent selection and S- (T and Q-) Weighting have been investigated in the literature, ranging from the extremely simple to complex techniques involving large-scale processing of the retrieval collection or of another resource. Selection methods which process the retrieval collection to extract collocation and/or co-occurrence information are the single most effective set of selection methods reported in the literature.

However, co-occurrence/collocation frequency calculation involves subjecting the retrieval collection to intense and onerous processing - something which may not always be feasible where the collection changes on a regular basis. In addition, little or no work has yet been carried out in the area of dictionary lookup. We maintain that dictionary characteristics, such as scale, coverage rate and entry definition, can have a significant affect on retrieval performance separate from that of the selection method employed. Furthermore, we contend that a judicious choice of dictionary based on a sound knowledge of dictionary characteristics can obviate the necessity for more complex equivalent selection methods requiring large-scale processing of the retrieval collection.

We consider the question of dictionary scale (the average number of equivalents per term listed in the dictionary) to be very important, as it has a direct effect on the number of equivalents to be considered at selection time. Our work on dictionary scale is presented in chapter 4. Following this, the issue of minor variations in query translation content having an effect on retrieval performance for dictionaries of similar scale obtained from similar sources, is dealt with in chapter 5. Here we examine how the micro-contents of dictionary entries affect the retrieval performance of associated query translations. Furthermore, we examine the issue of multiple occurrences of a single equivalent (*equivalent repetition*) in a dictionary entry. We shall see that such repetition, and the decision to include or exclude it from dictionary entries, can radically affect query translation retrieval performance for our retrieval system.

In chapter 6, we look at some pre-translation query modification methods which do not need a comparable document collection to be available in the source language, as this is yet another resource which may not always be available. This is in keeping with our focus on CLIR with few or no hand-crafted resources. Finally, also in chapter 6, we build on what we have learned in the previous two chapters to explore simple equivalent selection and S-weighting methods which rely on dictionary information

only. This should allow us to determine whether it is really worth putting so much effort into complex equivalent selection methods, or whether a carefully chosen dictionary combined with simple selection and S-weighting methods would be just as effective. Finally, we combine all of the insights obtained in carrying out this work to create a "best" set of query translations, and compared its retrieval performance with respect to the monolingual upper bound (see chapter 6). We have not done any work on post-translation query modification as a satisfactory method, post-translation blind relevance feedback, already exists. Chapter 7 concludes this thesis of our work with a summary of what has been achieved and gives some suggestions for how our ideas could be developed further.

Chapter 3

Experimental Environment

The previous chapter presented the state-of-the-art in CLIR research and explained why we concentrated on dictionary-based query translation in our work. This chapter describes the data, retrieval engine and experiment formats we employed and outlines how we obtained the CLIR dictionaries studied in subsequent chapters.

3.1 Experimental Data

In chapter 2, a detailed account was given of the methodology followed by participants in the TREC, CLEF and NTCIR CLIR evaluation initiatives. Our own experiments also employed this evaluation framework. We performed dictionary-based translation from English to French of the bag of words queries derived from a request set comprising the TREC-6, TREC-7 and TREC-8 English topic description fields, using a variety of *CLIR dictionaries* (machine-readable bilingual dictionaries suitable for use by a query translation system) and running our resulting query translations on a subset of the TREC-6 French document collection.

3.1.1 Query Set

Our query set was composed of the bag of words queries derived from a request set comprising the description fields of the 80 TREC-6, TREC-7 and TREC-8 English-language CLIR topics for which relevance judgements were available.[1]

This is a very small set of "data points" when compared with data sets in, for example,the field of machine learning, and therefore caution is required in generalising any of the conclusions reached in this thesis. However, a paucity of relevance information is one of the main problems associated with ALL experiments in IR, and in CLIR in particular. At the time this thesis was due to be submitted, a few hundred more queries with relevance data were available thanks to continuing TREC and CLEF evaluations, however, they were not incorporated into our experiments as we wanted to retain the same query set throughout.

The number of relevant documents in the collection for each query also varied quite considerably, with an average of 24 and a standard deviation of 26. With such a high standard deviation, the potential of a query with very few relevant documents to introduce artifacts into our results due to the query sampling methods employed in our experiments is non-negligible and is discussed in chapter 4. Unfortunately, as we shall see, it was difficult to see how this potential artifact could be removed without introducing another type of bias into our query sampling method, and so we retained our original method. This potential for bias must be borne in mind along with the other caveats regarding CLIR experiments in general expressed in chapter 2 when we consider our results.

We formed a request from a given TREC topic by extracting its description field. The description field of a TREC topic consists of a single sentence (see 2.2.1). For example, the description field of TREC cross language topic 3 is:

[1]The requests obtained from these TREC topics were similar in content and format and so it was reasonable to group them in a single test request set. Since evaluation data was not available for topic 25, we omitted it from our test request set completely, leaving 80 requests in total.

acupuncture:	acupuncture, acuponcture

Figure 3.1: Sample CLIR Entry

What measures are being taken to stem international drug traffic?

In processing an English-language topic description field (a request) to derive the bag of words query form, stopwords and punctuation were removed, all characters converted to lower case and the GATE lemmatiser (developed at the University of Sheffield as part of the GATE project [25]) applied to the remaining words. Our processed query 3 is:

measure stem international drug traffic

Lemmatisation and stemming were briefly discussed in chapter 2. We used Lemmatisation to ensure the best use of the dictionary information under study.

3.1.2 Document Collection

For our target-language document collection, we indexed a subset of the SDA French collection, around 160MB in size and containing approximately 93,000 documents, consisting of the 1988 and 1990 SDA French documents only. The full SDA French collection, employed in the TREC-6 CLIR evaluation, was not used due to practical constraints. Processing of this French collection prior to indexing proceeded in an identical manner to the English topic description fields (see above). A French stopword list was supplied by Cardie at Cornell University and the INALF's French-language lemmatiser used to lemmatise the documents (the lemmatiser was provided with their French-language implementation of the Brill tagger [93]). For details of why we chose lemmatisation as our term normalisation method, see the discussion of stemming versus lemmatisation in chapter 2.

3.1.3 Evaluation Data

We used the relevance judgements provided by NIST as part of the TREC CLIR task and NIST's `trec_eval` program to measure the retrieval performance of our results. Given a set of relevance assessments and ranked retrieval output for a set of requests, `trec_eval` calculates a range of measures of retrieval performance as described in Voorhees and Harman's TREC-7 overview paper [100]. We quote the two measures Average Precision (*AvP*) and Average Exact Precision (*R-Prec*) in our work. (The former is the most widely quoted measure in the literature as a single-figure assessment of performance). We also display the precision over the top 20 retrieved documents (*Document Cutoff 20* or *DC20*) where available. Values are expressed as a percentage as opposed to a number between 0 and 1. This is purely a matter of personal taste.

3.2 CLIR Dictionaries

Since, as explained below, entries in a CLIR dictionary are rather different from those in the conventional printed dictionaries from which our CLIR dictionaries are derived, we will use the expressions *dictionary entry* and *printed dictionary entry* to mean an entry in a printed dictionary and *CLIR entry* to denote an entry in a CLIR dictionary.

In section 2.1, we defined the concepts of *CLIR dictionary, CLIR entry, term* and *equivalent*. A CLIR dictionary is an alphabetical list of *terms* in the source language, where each term is associated with a list of equivalents in the target language. (An *equivalent* was defined as a possible translation of a source-language term). Each such term-equivalent list pair is called a CLIR *entry*. We explained in chapter 2 that CLIR entries could theoretically contain other information concerning the associated term, such as the part of speech of each equivalent. CLIR Entries in the CLIR dictionaries employed in our experiments do not contain any information regarding a term other than the associated list of equivalents itself as our experiments do not require the presence of any other information. Figure 3.1 displays a sample CLIR English-French entry for the term *acupuncture*.

action:	action, effet, acte, action, intrigue, action, moteur, proces, action, justice, mecanisme, marche, action, mecanique, combat, engagement, action, executer

Figure 3.2: Sample CLIR Entry

3.2.1 Creating Our CLIR Dictionaries

Our CLIR dictionaries were derived by hand from paper printed editions of standard bilingual English-French dictionaries aimed at language learners. Here, we describe how one might obtain a similar dictionary using such a printed dictionary as a source.

To derive a CLIR entry for a given term or lemma from a printed dictionary, one proceeds as follows. If the printed dictionary from which the CLIR dictionary is being derived does not contain a dictionary entry for the term in question, there is naturally no corresponding CLIR entry. Otherwise, the term will be assigned one, and only one, CLIR entry in the CLIR dictionary.

Dictionary entries frequently provide a great deal of information regarding the relevant source-language term other than possible translations, such as, for example, usage examples or translations of common phrases containing the term. Since we are interested in the equivalents only, we discard anything in the printed dictionary entry that is not an equivalent.

Multiple occurrences of a given equivalent, or *repetition*, are permitted within a term's CLIR entry. Such repetition is an important phenomenon when it is included in queries passed to our retrieval engine and is considered in chapter 4 and 5 where we shall be looking at CLIR dictionaries both with and without repetition in their CLIR entries. Figure 3.2 shows a sample CLIR entry from one of our CLIR dictionaries, containing several instances of repetition.

Multiple Sub-Entries in the Printed Dictionary

A dictionary entry may contain several sub-entries corresponding to different senses or parts of speech of the term, with a number of equivalents proposed for each sense or part of speech (see, for example, figure 3.3) [2] Alternatively, the printed dictionary entry may list all possible translations of all senses and parts of speech together (see figure 3.4). We form a single list of all of the equivalents provided in the printed dictionary entry, adding equivalents to this list in the order in which they appear in the printed dictionary entry. Where there are sub-entries, this equivalent list is formed by creating a sub-list of equivalents for each individual sub-entry in the same manner, and then concatenating these sub- lists in strict order of appearance of the sub-entries in the printed dictionary entry. (The order in which equivalents appear in the printed dictionary entry is held to be important).

Finally, we process the equivalent list in a similar manner to the target-language document collection, removing all stopwords and applying lemmatisation, for example, so that a query translation obtained using our CLIR dictionary can be run directly on the document collection. The source-language term and this processed equivalent list form a CLIR entry in our CLIR dictionary.

Multiple Printed Dictionary Entries

Some printed dictionaries contain several different entries for a single term, differentiating, for example, parts of speech (e.g. the verb *to action* v. the noun *action*), or homographs (e.g. *lead* as in *to lead an expedition* v. *lead* as in *lead pipe*). These cannot be distinguished from one another in a query at run-time as IR queries do not typically contain enough contextual information to allow accurate word-sense disambiguation to take place. Therefore, we do not distinguish between them in our CLIR dictionaries.

To combine multiple printed dictionary entries into a single CLIR entry, we obtain a list of equivalents from each such printed dictionary entry separately as described above and then concatenate these lists of equivalents in the order in which the printed dictionary entries appeared in the printed dictionary to form the full list of equivalents for that CLIR entry. Once more, we note that the order of appearance of equivalents in the printed dictionary is preserved as it is considered to be important (see chapter 6).

[2] The reader will note that the example shown is French-English, rather than English-French, this was a mistake on the part of the author when scanning the image - however, the image has been retained since the aim is to show the structure of a sample dictionary entry, and the layout of both parts of the Collins-Robert Unabridged dictionary is identical.

action [aksjɔ̃] 1 nf a (*acte*) action, act. **faire une bonne** ~ to do a good deed; ~ **audacieuse** act *ou* deed of daring, bold deed *ou* action; **vous avez commis là une mauvaise** ~ you've done something (very) wrong, you've behaved badly.

b (*activité*) action. **être en** ~ to be at work; **passer à l'**~ to take action; **le moment est venu de passer à l'**~ the time has come for action; (*Mil*) **passer à l'**~, **engager l'**~ to go into battle *ou* action; **entrer en** ~ *[troupes, canon]* to go into action; **mettre un plan en** ~ to put a plan into action; **le dispositif de sécurité se mit en** ~ the security device went off *ou* was set in action; *voir* **champ, feu**[1], **homme**.

c (*effet*) *[éléments naturels, loi, machine]* action; *[médicament]* action, effect. **ce médicament est sans** ~ this medicine is ineffective *ou* has no effect; **sous l'**~ **du gel** under the action of frost, through the agency of frost; **machine à double** ~ double-acting machine *ou* engine.

d (*initiative*) action. **engager une** ~ **commune** to take concerted action; **recourir à l'**~ **directe** to resort to *ou* have recourse to direct

sanitaire et sociale health and social services departments.

e *[pièce, film]* (*mouvement, péripéties*) action; (*intrigue*) plot. ~! action!; **l'**~ **se passe en Grèce** the action takes place in Greece; **film d'**~ action film; **roman d'**~ action-packed novel.

f (*Jur*) action (at law), lawsuit. ~ **juridique/civile** legal/civil action; *voir* **intenter**.

g (*Fin*) share. ~**s** shares, stock(s); ~ **ordinaire** ordinary share; ~**s nominatives/au porteur** registered/bearer shares; ~ **cotée** listed *ou* quoted share; ~ **à dividende prioritaire** preference share (*Brit*), preferred share (*US*); ~ **de chasse** hunting rights (*pl*); (*fig*) **ses** ~**s sont en hausse/baisse** things are looking up/are not looking so good for him; *voir* **société**.

h (*Mus*) *[piano]* action.

i (*Sport*) move.

2 comp ▶**action en diffamation** (*Jur*) libel action ▶**action d'éclat** dazzling *ou* brilliant feat *ou* deed ▶**action de grâce(s)** thanksgiving ▶**action revendicative** *[ouvriers]* industrial action (*NonC*); *[ménagères, étudiants]* protest (*NonC*) ▶**l'action sociale** social welfare.

Figure 3.3: Dictionary Entry in Collins-Robert Unabridged Dictionary - Multiple Sub-Entries

action [aksjɔ̃] *nf* (*gén*) action; (COMM) share; **une bonne** ~ a good deed; **actionnaire** *nm/f* shareholder; **actionner** *vt* (*mécanisme*) to activate; (*machine*) to operate

activer [aktive] *vt* to speed up; **s'**~ *vi*

Figure 3.4: Dictionary Entry in Collins Gem Pocket Dictionary - No Sub-Entries

action:	action, effet, acte, intrigue, moteur, proces, justice, mecanisme, marche, mecanique, combat, engagement, executer

Figure 3.5: Sample Entry with Repetition Removed

For example, the homograph *lead* has the meanings *to lead an expedition* and *an element of the periodic table*. Given a notional dictionary containing two separate dictionary entries for these two meanings, we proceed as follows. After processing the first dictionary entry, we obtain the equivalent list *mener, entrainer*. The second dictionary entry yields *plomb*. We concatenate these lists in the correct order to obtain *mener, entrainer, plomb*, and add the CLIR entry:

lead: *mener, entrainer, plomb*

to the corresponding CLIR dictionary.

3.2.2 Our CLIR Dictionaries

We obtained three English-French CLIR dictionaries from paper printed editions of standard bilingual dictionaries (electronic versions were not available). These CLIR dictionaries, which we called *Large, SGem* and *VerySmall*, were derived from the Collins-Robert Unabridged, Collins Gem Pocket and Langenscheidt Lilliput English-French printed dictionaries respectively. The Collins-Robert Unabridged dictionary is a university-level dictionary which aims to be as complete as possible. The Collins Gem Pocket Dictionary, on the other hand, is a pocket dictionary of the type commonly used by GCSE-level students, but is certainly not a complete guide to English-French translation. Finally, the Lilliput dictionary was a tiny, one inch by half an inch "baby" dictionary that one might carry around in a handbag or day rucksack. We created CLIR entries in these CLIR dictionaries for the 385 terms in our test query set only.[3]

We also created further versions of two of our CLIR dictionaries, *LargeNoRep* and *SGemNoRep*, with all *equivalent repetition* removed from their CLIR entries. Repetition, as stated above, is when an equivalent appears more than once in a single term's CLIR entry. Repetition is removed by deleting from each CLIR entry's equivalent list any second or subsequent occurrence of an equivalent in that CLIR entry. For example, if we remove all repetition from the CLIR entry depicted in figure 3.2, we obtain that displayed in figure 3.5. We did this as repetition in CLIR entries is an important phenomenon affecting retrieval performance and is investigated in chapters 4 and 5. (We note that this type of repetition is distinct from an equivalent being repeated in a query translation due to it being provided as a potential translation for more than one query term. Our work does not address this type of equivalent repetition, only that due to repeated equivalents within a single query term's equivalent list).

There was no repetition in *VerySmall*, so it was not necessary to produce another version of it. An additional CLIR dictionary, *Teensy*, was derived from *VerySmall* automatically by copying the first equivalent only from each CLIR entry in *VerySmall* to form the corresponding equivalent list in *Teensy*. This was done as it was felt that *VerySmall* was too similar to *SGemNoRep* for the purposes of the dictionary scale experiments discussed in chapter 4.

For example, figure 3.6 shows how the *Teensy* CLIR entry:

abuse:*abus*

was derived from the *VerySmall* CLIR entry:

abuse:*abus, insulte, abuser*

This set of CLIR dictionaries is referred to in subsequent chapters as our *print-derived CLIR dictionaries*. This name is slightly misleading with respect to *Teensy* as it was not derived from an independent source, nevertheless we find it useful to label these CLIR dictionaries in this way.

[3]Thus no copyright infringement occurred.

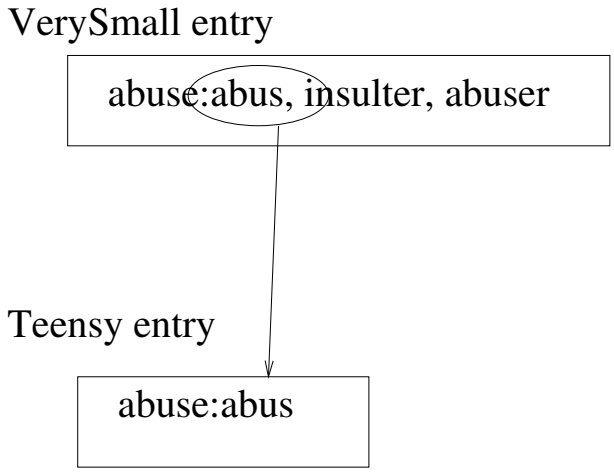

Figure 3.6: Deriving a *Teensy* CLIR Entry from a *VerySmall* CLIR Entry

Coverage Issues

Lack of coverage in the CLIR dictionary can have an effect on the retrieval performance of query translations [47], and we wanted to ensure that coverage issues did not introduce artifacts into our experiments. To ensure a coverage rate of 100%, CLIR entries were added by hand to *Large* and *LargeNoRep* for any of the 385 terms in our test query set which did not have a CLIR entry in the Collins-Robert Unabridged dictionary and which consequently were not assigned a CLIR entry in *Large* and *LargeNoRep* using the dictionary derivation method described above. For example, *Large* and *LargeNoRep* did not contain a CLIR entry for the proper noun *Waldheim*, so the CLIR entry:

waldheim:*waldheim*

was added to both. Most of the twenty or so CLIR entries added to *Large* and *LargeNoRep* in this way were proper nouns or identities, although there were a few that were not identical in English and French.

Instead of adding CLIR entries by hand to the other CLIR dictionaries, we applied a process called *coverage compensation*, which involves copying CLIR entries from *LargeNoRep* into the CLIR dictionary under consideration for those terms in our query set which did not have a CLIR entry in that CLIR dictionary. For example, *Teensy* did not contain a CLIR entry for *European*, so the CLIR entry:

european:*europeen*

was copied from *LargeNoRep* and added to *Teensy* (see figure 3.7). This is to avoid any artifacts being introduced into our experiments by lack of coverage. We copied approximately 60 such CLIR entries into *SGemNoRep* and approximately 120 into *VerySmall* and *Teensy*. We note that the entries copied in this way were not altered in any way - so if they contained more than one equivalent, we did not remove the second or subsequent equivalents, meaning that about 60 of the new entries in *Teensy* contained more than one equivalent.

Our Automatically-Derived CLIR Dictionaries

Two additional CLIR dictionaries, *AutoMediumNoRep* and
AutoVerySmall, were derived semi-automatically from *LargeNoRep*. This was done as we wanted to compare the behaviour of CLIR dictionaries of differing *scale* which were proper subsets of one another with those obtained from different printed sources (see chapter 4 for a definition of dictionary scale). *AutoMediumNoRep* was created by taking the first three equivalents from the first three senses of each term listed in *LargeNoRep* (see figure 3.8). The sense information was obtained by consulting the relevant printed dictionary by hand, as this information was not provided by the corresponding CLIR dictionary. Where a given term had fewer than three senses, the first three equivalents of every sense were copied, and where there were fewer than three equivalents for a given sense of a term, all equivalents for that sense were copied. Any repetition in the new CLIR dictionary's CLIR entries was then removed.

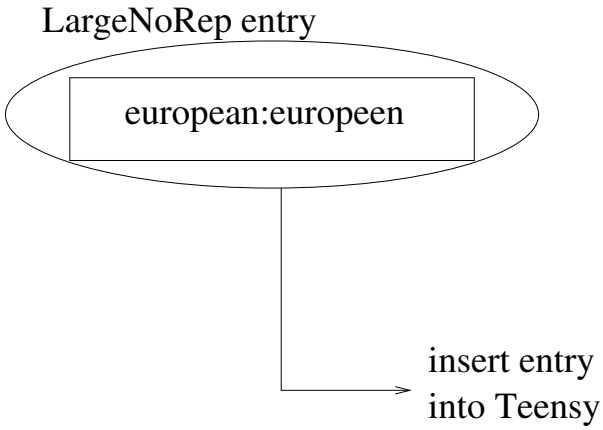

Figure 3.7: Example of Coverage Compensation

Figure 3.8: Deriving an *AutoMediumNoRep* Entry from *LargeNoRep*

AutoVerySmall was obtained by taking the first equivalent from each term's equivalent list in *LargeNoRep* to create its entries (see figure 3.9). Both of these new CLIR dictionaries had the same coverage level as *LargeNoRep*, namely 100%. We term these CLIR dictionaries our *automatically-derived CLIR dictionaries* to differentiate them from the print-derived dictionaries discussed above.

3.2.3 Translating a Query Using a CLIR Dictionary

To obtain the Add-All-Equivalents translation (see section 2.14.2) of a source-language query, for each source-language query term, we take the entire equivalent list given in its CLIR entry and add all of the equivalents in this list to the query translation. If there is repetition in a given term's equivalent list, all occurrences of the repeated equivalent are added to the query translation. Thus, the final query translation consists of all of the equivalents provided by the relevant CLIR dictionary for all of the terms in the original source-language query. This is the Add-All-Equivalents query translation method discussed in chapter 2. It is employed in most of our experiments using dictionaries without repetition in their entries as a baseline for performance.

3.3 Information Retrieval Engine

We employed an information retrieval engine developed at the University of Cambridge Computer Laboratory by Pierre Jourlin. This engine applies the probabilistic model of information retrieval [94, 81], where the potential relevance of a document to a query is calculated as a conditional probability and is implemented via term weights [94].

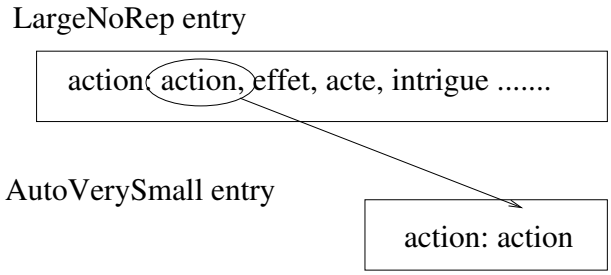

Figure 3.9: Deriving an *AutoVerySmall* Entry from *LargeNoRep*

Jourlin's system also incorporated many advanced retrieval features, such as relevance feedback and query expansion. However, we disabled these features for our experiments as we wanted to observe the behaviour of individual dictionary characteristics and S-weighting methods in isolation. This means that the absolute performance values reported in subsequent chapters will be lower than what one would normally expect from this retrieval engine.

Each retrieval run in our experiments was carried out twice - once using the standard retrieval engine term weighting mechanism as described above *(R-weighting enabled)*, and once where this weighting formula was replaced by a simple term-counting method where 1.0 was added to the score of a document for each occurrence of a query term in that document *(R-weighting disabled)*. This was to ensure that any results observed were not an artifact of the term weighting scheme of this particular retrieval engine. In both cases, the total score for a given document is the sum of the relevant weights for the query terms found within it.

The retrieval engine does not impose any structure on queries submitted to it. This means that no equivalent grouping is possible. It processes an Add-All-Equivalents translation, for example, as an unstructured list of independent terms, and calculates a separate retrieval engine weight for each. If duplicate terms are present in a given query translation, no attempt is made to correlate them, they are simply treated as different terms. Therefore, repeating a term in a query input to this engine has the same effect as doubling its weight in the query. It is also possible to specify a multiplicative S-weight for a query term by specifying it immediately in front of the relevant term in the query text. The engine looks for such an S-weight in front of each term, and if one is not found, the default S-weight 1.0 is applied.

This query processing structure influences the manner in which our query translation experiments were constructed. If our retrieval engine handled queries differently, we would have to structure our query translations differently - for instance, explicitly disabling any automatic equivalent grouping or query expansion facilities. Persons wishing to replicate the experiments reported here must adjust their query translation algorithms to reflect the manner in which queries are processed by their retrieval system.

3.4 Difference Runs

We developed a technique called *difference runs* which enables us to compare two different French translations of a given English query and to ascertain which elements of these two translations lead to a difference in performance between these two translations when the translations were run on the French document collection.

Let us say that we have two translations $T1$ and $T2$ of a given query Q, with different retrieval performance scores. We want to find out what differences between $T1$ and $T2$ were responsible for this observed performance variation.

We define a *point of difference* as an operation which, if carried out on $T1$, would make $T1$ more similar to $T2$, by adding, deleting or repeating a single equivalent in $T1$. (This is similar to string editing [48]).

We then enumerate the points of difference between $T1$ and $T2$. For example, let us say we have the two translations :

$$T1 = \{a, b, b, c, f, g\}$$
$$T2 = \{a, b, d, e, f, g\}$$

where a, b, c, d, e, f and g are translation equivalents.
The four points of difference between $T1$ and $T2$ here are:

- deletion of repetition of b in $T1$
- deletion of occurrence of c in $T1$
- addition of d, to $T1$
- addition of e, to $T1$

Now, we form a set of $T1_{1...n}$ new queries, where n is the number of points of difference between $T1$ and $T2$. Each $T1_i$ is constructed by applying a single point of difference operation to $T1$, for example, deleting b from $T1$. For the above example, we obtain the set:

$$T1_{1...n} = \{\{a, b, c, f, g\}, \{a, b, b, f, g\}, \{a, b, b, c, f, g, d\}, \{a, b, b, c, f, g, e\}\}$$

We then perform retrieval using each of these $T1_i$ and note the result. If the performance of a given $T1_i$ is the same as that obtained for $T1$, we conclude that the point of difference embodied in $T1_i$ is not responsible for the difference in performance between $T1$ and $T2$. If, on the other hand, there is a difference between the performance obtained by $T1_i$ and $T1$, (where the performance recorded for $T1_i$ is less than or greater than that recorded for $T1$), we conclude that the point of difference embodied in query $T1_i$ plays a role in the overall difference in performance between $T1$ and $T2$. We note that it is perfectly possible for some points of difference to result in a drop in performance, although $T2$ performs better overall than $T1$, and vice versa - as we shall see in subsequent chapters, one use of Difference Runs is to find out which equivalents in a translation contributed to the overall result and which detracted from it.

In this way, we can determine which equivalents in the two translations are responsible for performance differences, and which by their presence or absence play a role in improving or degrading performance. This technique assumes that the equivalents in a query translation are independent of one another, which is not strictly correct but is a commonly-made assumption in information retrieval.

Figure 3.10 displays a sample output of this technique. First of all, the query number, 58, is given. Then, we are told that the *Human* and *Perfect* translations (see chapter 4) are being compared, with R-weighting disabled (*Unweighted*). The R-Prec of the Human and Perfect translations of query 58 are 33% and 50% respectively. The points of difference being assessed are:

- adding *baisse* to the Perfect translation
- adding *cote* to the Perfect translation
- adding *devoir* to the Perfect translation
- adding *autour* to the Perfect translation
- deleting *littoral* from the Perfect translation
- deleting *baisser* from the Perfect translation
- deleting *considerablement* from the Perfect translation

We can see that adding *baisse* caused performance to drop. We concluded that the presence of this equivalent in the Human translation was responsible for the observed difference in performance between the two translations. As a full set of difference run results for many different translation pairs takes up a great deal of space, we generally include them as a appendix, giving a summary in our text. Since this technique assumes that equivalents in a query translation are independent of one another, and this assumption is incorrect, this technique is a bit "rough round the edges" - there will be factors resulting from equivalent interdependence which can also influence retrieval performance which this technique does not measure. Therefore, we deem any observed points of difference to be partially responsible for any drop/rise in performance observed, rather than wholly.

3.5 Significance Testing

In our experiments, we compared the performance of two alternative sets of translations of our 80 test queries set by examining the average values of AvP and RP in both weighted and unweighted cases. This gave us four (or six) points of comparison for a given pair of runs (sets of query translations). Where the average value of a given metric, for example, AvP in the Weighted case, for two runs were different, we wanted to know if this difference was due to an underlying tendency of one of the translation methods to produce query translations that tended to perform better.

We eliminated the effect of variations in the absolute performance values to concentrate on the degree of difference as follows. For each pair of translations whose performance we want to compare, we construct a new representation of the query-by-query results for each of the 4 metrics in question (AvP and R-Prec in the unweighted and weighted cases).

```
Query 58

Run              Unweighted
--------------------------
Human                    33
Perfect                  50
--------------------------
Adding to Perfect
--------------------------
baisse                   33
cote                     50
devoir                   50
autour                   50
--------------------------
Removing from Perfect
--------------------------
littoral                 50
baisser                  50
considerablement         50
--------------------------
```

Figure 3.10: Sample Output of Difference Runs Technique

If the value quoted for a given query and a given metric in the first set of query translation performance values is greater than the second, we allocate the value 1 to the first query translation and 0 to the second in our new representation, and vice versa. Where both values are equivalent (within 1% of one another) 1 is allocated to both query translations. This is done for each of the pairs of query translation performance results corresponding to the 80 queries in our test set.

Then, a two-tailed paired T-Test is carried out on this new representation to see if the set of query translations which scored better on average for a given metric obtained a superior score because a different, better result is recorded in a significant number of cases, or if this was due to chance only. A separate T-Test was carried out for AvP and R-Prec in both the Unweighted and Weighted cases, resulting in 4 T-Tests per comparison in total. The null hypothesis in every case is that the differing average performance values observed for a given pair of sets of query translations is due to chance and not to a significant difference in the underlying performance behaviour.

A table giving the probability of the null hypothesis in each case is displayed in our results wherever such tests have been carried out.

For example, in figure 3.11 we have a fragment of the Weighted AvP results for the French Human and Perfect Dictionary translations of queries 1 to 12 (see chapter 4). The new representation is shown, along with the probability of the null hypothesis for Weighted AvP taking into consideration only these 12 queries and their results. We see that with a probability of 1.0 the null hypothesis is strongly supported, indicating that any underlying difference observed in the new representation in figure 3.11 is not significant.

All significance tests mentioned in subsequent chapters were of the type described above unless explicitly stated otherwise. (In chapter 6, we also employed the Wilcoxon and Sign tests on the raw data for selected results, as the test discussed above was felt subsequently to have been too conservative).

3.6 Conclusions

In this chapter, we discussed the data and evaluation framework we used to carry out the experiments reported in subsequent chapters. We outlined how a CLIR dictionary may be derived from a printed dictionary and detailed the various CLIR dictionaries we created for our experiments. We also described the retrieval engine we employed. We gave an account of our *difference runs* technique, an analysis tool that enables us to determine which aspects of two different translations of the same query are responsible

QNum	French Human	Perfect	FH New	Perfect New
2001	14	10	1	0
2002	7	30	0	1
2003	8	15	0	1
2004	21	5	1	0
2005	20	100	0	1
2006	28	51	0	1
2007	20	7	1	0
2008	32	41	0	1
2009	44	14	1	0
2010	47	38	1	0
2011	6	6	1	1
2012	13	23	0	1

Figure 3.11: Fragment of New Representation for Significance Testing, French Human v. Perfect Dictionary Translations

Runs Compared	W AvP
French Human v. Perfect Queries 1-12	1.0

Figure 3.12: Significance Test Results for the New Representation of Queries 1-12, Comparing French Human with Perfect Dictionary Translations

for any observed difference in retrieval performance between them and which we have used heavily in our experiments. Finally, we outlined how we performed the significance tests we applied to compare the results of different sets of translations of our test query set.

The next chapter introduces the first stage of our own work, including our experiments on the effect of dictionary scale on the retrieval performance for query translations.

Chapter 4

Dictionary Scale in Query Translation

In chapter 2, we discussed our interest in CLIR methods which could still be employed when hand-crafted resources such as MT engines and aligned parallel corpora were not available for the language pair and/or subject domain under consideration. We saw that simple bilingual dictionary-based bag of words query translation methods, although not always as effective as, for example, MT-based request translation, had the widest range of applicability. In addition, we saw that quite good levels of performance have been reported in the literature for dictionary-based approaches. In chapter 3, we gave an example of how a simple bilingual CLIR dictionary could be derived from a standard printed bilingual dictionary, either automatically or by hand, and described a number of such dictionaries we obtained in this way.

Returning to chapter 2, we divided the process of dictionary-based query translation into four logical stages:

1. Pre-translation query modification
2. Dictionary lookup
3. Equivalent selection and T-weighting
4. Post translation query translation modification

We noted that the majority of research had focused on the third step. We contend that the previous two steps are at least as deserving of attention as equivalent selection. In particular, we propose that careful examination of dictionary and query characteristics affecting these two stages can lead to insights which, when applied to the process of query translation, result in a significant improvement in retrieval performance for translated queries almost as good as that achieved by implementing the more complex, involved equivalent selection strategies discussed in the literature.

In this chapter, we look at the issue of dictionary *scale*, where scale is defined as the average number of distinct equivalents listed in a given CLIR dictionary per entry. The scale of a dictionary has a considerable influence on the number of equivalents proposed for inclusion in the query translation. A smaller scale dictionary will provide fewer equivalents on average, thus making the task of any equivalent selection, T-Weighting or Q-weighting modules easier. However, a query translation obtained using such a dictionary may also suffer from what we call the *crucial equivalent effect* - where the absence of one or two important equivalents from the entries of the dictionary lead to reduced retrieval performance despite the reduced number of equivalents proposed by the dictionary. The work in this chapter investigates how these two factors - the presence of fewer equivalents and the higher risk of missing a crucial one - interact with respect to retrieval performance for dictionaries of differing scale. We also discuss some preliminary work on the subject of the effect on retrieval performance of the interaction between our retrieval system's query processing methods and equivalent repetition in a single term's equivalent list in query translations, which is dealt with more fully in chapter 5.

Work proceeded by our posing a sequence of (not always closely related) hypotheses, and is therefore presented in this manner in this and subsequent chapters. Once each hypothesis was either verified or

disproved, we moved on to the next. For convenience, we have numbered our hypotheses according to chapter and according to their order of appearance in a given chapter.

We began with some control runs, and showed that query translations are very sensitive to small variations in query translation content. We then went on to look at dictionary scale. We returned to the conclusions of our control runs with a look at how the crucial equivalent effect and the swamping effect interact for translations obtained from dictionaries of differing scale. Finally, we look at a possible way of combining several dictionaries to reduce the crucial equivalent effect, and present some preliminary work on the role of equivalent repetition within a single term's equivalent list and its effect on retrieval performance.

The next section contains some remarks on the metrics used for evaluating our runs and the presentation of results in this and subsequent chapters.

4.1 Presentation of Retrieval Run Results

As stated in chapter 3, all experiments presented in this chapter were carried out twice - once with retrieval engine R-weighting enabled (*Weighted* runs), and once with it disabled (*Unweighted* runs). This was to ensure that any artifacts introduced into our results by the R-weighting scheme implemented by this particular retrieval engine could be detected. However, in the main, general trends noted for Weighted runs were reflected in the corresponding Unweighted results.

We noted in chapter 2 that *Average Precision*, or *AvP* was the most common performance metric quoted in the literature. This is because it provides an assessment that ranges over the entire retrieved document set of 1000 documents. *Average Exact Precision*, or *R-Prec*, on the other hand, looks at the top M documents only, where M is the point in the retrieved document list where recall is equal to precision. In our results, we present the values calculated for AvP, R-Prec and also precision over the top twenty retrieved documents (*Document Cutoff 20* or *DC20*). The aim here is to give a fairly wide-ranging basis for comparing results. For our Difference Runs technique, we used R-Prec only, as employing all three measures would have been unwieldy and complicated. The choice of R-Prec here over the other two metrics was a matter of personal taste.

In addition, results are quoted as percentage values rather than numbers between 0 and 1, with rounding to two figures. Therefore, R-Prec of 0.2453, for example, would be displayed as 25%. The percent sign, where not shown, is implied. Finally, the reader is reminded that performance values for all metrics quoted here will be artificially low, as we have disabled many features of the retrieval engine which a system participating in TREC, for example, would have enabled. We did this to ensure that we could observe the effect on retrieval performance of individual factors in a strictly controlled environment.

4.2 Control Experiments

Before carrying out the experiments outlined above, we did some control experiments to set an approximate upper bound on performance. It would not be fair to expect any of our query translations to perform better than this approximate upper bound.

As stated in chapter 3, we obtained 80 French and 80 English queries by processing the description field of the English and French versions of the TREC-6, TREC-7 and TREC-8 CLIR track topics. The French queries were lemmatised using the lemmatiser supplied with the INALF's French version of the Brill tagger [93] and the English queries with the University of Sheffield's GATE lemmatiser [25]. This set of 80 queries constitutes a small sample - as stated in chapter 3, paucity of relevance data has been one of the major stumbling blocks in IR research, although the advent of TREC has done quiet a lot to improve this situation. However, at the time where our experiments were begun, relevant data was available only for these 80 queries, and so we retained this data set throughout our experiments. We did not add additional queries as they became available with subsequent TREC and CLEF evaluations due to results reported in the literature which indicated that results for the same method and retrieval collection could vary considerably over different collections and queries [44].

Run	UnWRP	WRP	UnWDC20	WDC20	UnWAvP	WAvP
French Human	18	34	18	28	17	33
Perfect Dictionary	16	31	14	25	15	31

Figure 4.1: Control Runs

Runs Compared	UnW RP	UnW AvP	W RP	W AvP
French Human v. *Perfect Dictionary*	0.2	0.175	0.012	0.009

Figure 4.2: Significance Test Results - Probability of Null Hypothesis

4.2.1 French Human Queries

First, we ran these French queries on the French document collection. This run was entitled the *French Human* run, as the TREC topics from which the queries were derived were either translated from other languages into French by a human translator or were created by a French-speaking human in the first place [89, 17, 13]. The results, showing AvP, R-Prec and DC20, are displayed in Figure 4.1.

4.2.2 *Perfect Dictionary* Translations

The French queries, although equivalent in content to their English counterparts, are not exact word-by-word translations of them. For example, the English description field of topic 18 is:

Is perfume one of the most inflation proof luxury items in the world?

whereas the corresponding French sentence is:

Pourquoi le parfum est-il un des produits de luxe le moins affecte par l'inflation?

As a result, it is unlikely that identical performance would always be recorded for a word-by-word translation of a given English query would and the corresponding French Human query. (We have disabled phrase discovery in queries, so all our translations will be word-by-word). Therefore, we also created a set of word-by-word translations of the English queries called our *Perfect Dictionary* translations, by selecting by hand the "best" equivalent for each source-language query term from the equivalents proposed in *LargeNoRep*. The "best" equivalent for a term was selected by us based on our (comprehensive) knowledge of French and familiarity with the contents of the original topics. The *Perfect Dictionary* translations represent a theoretical maximum performance for any automatic word-by-word query translation method. Retrieval results for the *Perfect Dictionary* translations are displayed in Figure 4.1, where we can see that they are slightly inferior to those recorded for the French Human queries for all three measures. The differences in results between the respective Unweighted runs were not significant according to the significance testing methodology set out in chapter 3, but the Weighted runs were significant to 95% (see Figure 4.2).

4.2.3 Performance Variations

A query-by-query comparison of R-Prec performance for the French Human queries and the *Perfect Dictionary* translations yielded some interesting results. (We compare R-Prec results as these observations will form the basis for a set of Difference Runs). With R-weighting disabled, there were 20 queries for which the *Perfect Dictionary* translation obtained better R-Prec performance than the corresponding French Human query. With R-weighting enabled, there were 47 queries in our test set of 80 queries for which R-Prec results differed, with superior performance for the *Perfect Dictionary* translation recorded for 14 of these. We hypothesised that although the two sets of query "translations" (viewing the French Human queries as a set of "translations" of the English queries as well) were very similar in composition, minor variations in content resulted in quite large differences in performance for a certain number of queries. We posed the following hypothesis:

4.3 Hypothesis 4A: Retrieval Performance of Query Translations is Very Sensitive to Small Variations in Composition

To verify the hypothesis that an observed difference in performance between a given French Human query and the corresponding *Perfect Dictionary* translation was due to small, and not major, differences in query translation composition, we employed our Difference Runs technique as described in chapter 3. We would expect to see that one, or at the most two or three, points of difference would be found to have affected R-Prec performance.

4.3.1 Verifying Hypothesis 4A

Since performing Difference Runs for all 80 queries in our test set would have been too time-consuming, we selected a random sample of queries to analyse for the purposes of investigating this hypothesis. The sample size in each case was roughly a quarter to a sixth of the size of the set from which it was selected. The author considers this sample size to be large enough to be representative of the related query sets as a whole.

Due to the large standard deviation (26) from the average number of relevant documents (24) per query in the collection, there is a possibility that any selection of a sample set of queries could result in the accidental choice of queries with a very small number of relevant documents only, thus introducing an unintentional bias into our results. However, deliberately selecting queries for inclusion in a sample set based on having a number of relevant documents in the collection similar to the mean, or greater than the mean would also introduce bias, albeit in the other direction. Therefore, we chose our sampling method (random number generation) in such a way as as to ensure that the number of documents relevant to a particular query did not influence our choice of sample sets. There is a potential for artifacts as stated above, however, as we shall see, we observed similar results over several experiments and query samples, and so it is unlikely that our major conclusions were due to bias in our samples.

Five queries were selected at random from the set of 20 queries for which the *Perfect Dictionary* translation achieved the best R-Prec performance with R-weighting disabled - queries 12, 58, 66, 69 and 81 - and 5 queries for which similar results were recorded with R-weighting enabled - queries 3, 5, 14, 43, and 48. We performed a set of Difference Runs between these queries' *Perfect Dictionary* translations and the corresponding French Human queries. The full details of these Difference Runs are to be found in Appendix 4.(i).

We carried out separate sets of difference runs for the Unweighted and Weighted results as we wanted to see if the same effect was observed in both cases. Were this not the case, we would need to think considered whether any observed performance differences were due to the nuances of the retrieval engine R-weighting strategy only. Furthermore, if the hypothesis to be confirmed is important, we would expect it to be confirmed in both cases. As it happened, both here and in subsequent experiments, the conclusions reached for the Weighted results mirrored those of the Unweighted results, but it was not guaranteed at the outset that this would be so, and therefore we needed to check this for each set of experiments.

Hypothesis 4A was confirmed. In many cases, the R-Prec performance difference was due to the presence of a single equivalent in the *Perfect Dictionary* translation that did not occur in its French Human version, or the converse. For example, for query 58 (also presented in chapter 3 where we explained our Difference Runs technique), the presence of the equivalent *baisse* in the French Human query was the sole reason for the lower performance of the latter. There were also some queries for whom the lower R-Prec performance of the French Human version was the result of a combination of the effect of two or three equivalents. For example, for query 3, the removal of the equivalents *international, drogue* and *trafic* from the *Perfect Dictionary* translation resulted in a drop in performance. We concluded that the R-Prec performance differences observed were due to a small number of points of difference between the two versions of the examined query in each case, verifying our hypothesis.

4.3.2 Concluding Remarks on Hypothesis 4A

This means that retrieval performance will be sensitive to both small gaps in the translation system's knowledge and to minor translation errors. In addition, this implies that two different human-generated translations of the same query will not necessarily obtain the same retrieval performance, a conclusion already reached by Sakai [82]. This would indicate that a combination of translations from different sources would work best, eliminating any drops in performance caused by gaps in the knowledge of a given translation system or by errors in translation. This finding was be taken into account when considering our which approach to take to various problems below and in subsequent chapters.

From there, we moved on to our first translation task, producing query translations using our CLIR dictionaries (see chapter 3) and Add-All-Equivalents dictionary lookup (see chapter 2).

4.4 Hypothesis 4B: Translations Obtained Using Smaller Scale Dictionaries Perform Better

We defined dictionary *scale* in chapter 3 as the average number of distinct equivalents proposed by the CLIR dictionary per query term. Since query translations obtained using a smaller scale dictionary will contain fewer equivalents, we hypothesised that they would perform better than translations derived using a larger-scale dictionary. In addition, we hypothesised that there would be a rough inverse correlation between dictionary scale and retrieval performance, due to what we call the *swamping effect*. This is the most important hypothesis posed in this chapter.

4.4.1 Significance of Hypothesis 4B

If we were to demonstrate that a very small-scale dictionary was the best to use for translating queries, it would reduce considerably the time and effort needed to convert electronic versions of printed dictionaries to a format that can be employed by CLIR software, as smaller-scale dictionaries tend to be derived from sources with a much simpler internal structure providing less information, for example, a printed pocket mini-dictionary.

Many of the CLIR dictionaries currently employed by CLIR researchers have been derived from electronic bilingual dictionary files not at all suited to CLIR. Hull and Grefenstette [47] discuss the considerable amount of work they needed to carry out to transform electronic versions of the Hachette French-English dictionary to a format they could use. Since the printed dictionaries from which our smaller-scale dictionaries were derived tend to be much smaller in size, contain less information in their entries and do not have anything like the degree of complexity in the way information is presented, the effort required at this stage would be considerably reduced.

Although recourse to a more comprehensive printed dictionary would still be necessary where *coverage compensation* - necessary for a high rate of coverage - (see chapter 3) was to be applied, this still represents a considerable reduction in the time taken to process a language learner's dictionary to produce a CLIR dictionary, as only the printed dictionary entries corresponding to terms missing from the small-scale printed dictionary would have to examined. Consequently verifying this hypothesis would be useful for increasing the portability and decreasing the cost of developing dictionary-based CLIR systems.

4.4.2 Verifying Hypothesis 4B

We obtained a set of French translations of our 80 English queries from each of our print-derived and automatically-derived dictionaries (see chapter 3) using the Add-All-Equivalents selection method (see chapter 2). Our version of this method performs no selection or T-Weighting after lookup. No pre- or post-translation query modification was implemented. The aim was to observe the effect of varying dictionary scale in isolation. We note that this type of experiment can also result in the importance of dictionary scale being exaggerated, as T-weighting, post-translation query modification and so on could nullify the differences in performance between query translations obtained from dictionaries of differing scale. Nevertheless, we felt this was preferable to having several factors influencing results and then trying to find out which factor was responsible for any observed performance effects.

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
LargeNoRep	5	13	6	12	5	15
SGemNoRep	9	21	10	19	10	22
VerySmall	9	21	10	18	11	23
Teensy	11	24	11	19	12	25
AutoMediumNoRep	8	18	8	17	10	21
AutoVerySmall	12	25	11	21	13	26

Figure 4.3: Results for Add-all-equivalents Runs with Dictionaries of Differing Scale

Runs Compared	UnW RP	UnW AvP	W RP	W AvP
LargeNoRep v. *SGemNoRep*	0.0	0.0	0.0	0.0
LargeNoRep v. *VerySmall*	0.01	0.0	0.0	0.0
LargeNoRep v. *Teensy*	0.01	0.0	0.0	0.0
SGemNoRep v. *Teensy*	0.54	0.191	0.330	0.153
VerySmall v. *Teensy*	0.08	0.01	0.131	0.000
LargeNoRep v. *AutoMediumNoRep*	0.01	0.0	0.0	0.0
LargeNoRep v. *AutoVerySmall*	0.0	0.0	0.0	0.0
AutoMediumNoRep v. *AutoVerySmall*	0.005	0.0	0.003	0.0

Figure 4.4: Significance Testing for Dictionary Add-All-Equivalents Translations, Probability of Null Hypothesis

Since our dictionaries contained entries for the 385 terms in our 80 English queries only, our scale measurements were calculated using solely these terms (scale is the average number of equivalents provided by the dictionary per term). Scale was measured after coverage compensation had been applied to *SGemNoRep, VerySmall* and *Teensy*. We obtained scale values of 6.0, 2.5, 1.8 and 1.0 for the print-derived dictionaries *LargeNoRep*, *SGemNoRep*, *VerySmall* and *Teensy* respectively and values of 3.2 and 1.0 for the automatically-derived dictionaries *AutoMediumNoRep* and *AutoVerySmall*. The results of running these sets of translations on the document collection are displayed in figure 4.3. Since coverage is not an issue, we can be confident that dictionary scale is the main factor influencing these results. Once more, results are stated in terms of AvP, R-Prec and DC20.

Looking at figure 4.3, we see that for all three performance metrics, and for both Weighted and Unweighted runs, dictionary scale and retrieval performance appear to be inversely related. The translations obtained using the largest-scale dictionary, *LargeNoRep*, got the lowest scores in all cases, and the translations obtained from the smallest-scale dictionaries, *Teensy* and *AutoVerySmall*, the best, with the others somewhere in between.

Significance testing (see figure 4.4) showed that for both measures tested (AvP and R-Prec), in both the Unweighted and Weighted cases, the results of the *LargeNoRep* translations were significantly different from those for all other dictionaries' translations quoted in figure 4.3. In addition, the difference in average performance between the *AutoMediumNoRep* and the *AutoVerySmall* translations was significant in all four cases. The differing results of the *SGemNoRep* and *Teensy* translations were not significant. AvP results for the *VerySmall* compared to the *Teensy* translations were significant, the corresponding R-Prec results were not. Consequently, caution must be exercised when comparing the performance of translations obtained using *SGemNoRep, VerySmall* and *Teensy*.

In addition, comparing R-Prec for the smallest-scale dictionaries' translations and our control runs, *Teensy* and *AutoVerySmall* obtained 67% and 72% of the corresponding monolingual performance (the

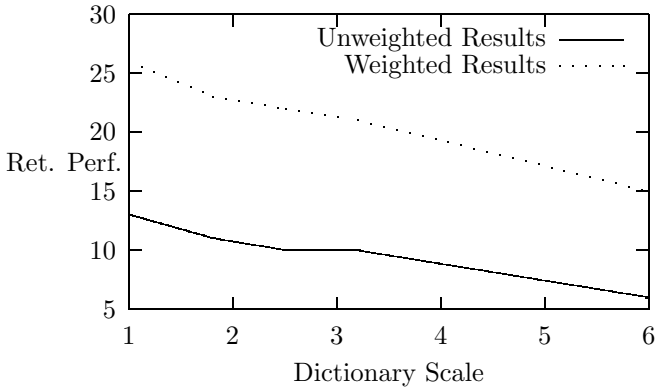

Figure 4.5: Plot of Dictionary Scale Against Retrieval Performance

French Human queries), respectively with R-weighting disabled, and 74% and 76% with it enabled. This corresponded to 75% and 81% of the *Perfect Dictionary* translation performance respectively with R-weighting disabled, and 81% and 84% with it enabled. This constitutes a respectable level of performance for CLIR without any further processing (Ballesteros and Croft quote results like 85% of monolingual performance [10]), and compares favourably with many of the results quoted in our literature survey (see chapter 2), although the usual caveats apply to any direct comparison (see section 2.4). Hypothesis 4B has been verified. This has important implications for the portability of dictionary-based CLIR, as discussed above.

The graph in figure 4.5 plots, for each of our auto-derived and print-derived dictionaries, dictionary scale against the performance (R-Prec) of the related query translations, in both Weighted and Unweighted cases. We can see that dictionary scale is roughly inversely proportional to query translation performance for the dictionaries that we employed in our experiments. However, we saw above that the differences in performance between three of the smaller-scale dictionaries' translations was not significant. Further experiments need to be carried out using many different CLIR dictionaries to further support this conclusion, however, this was beyond the scope of our experiments.

4.5 Hypothesis 4C: The *Swamping Effect* is the Cause of This Apparent Inverse Proportionality

We hypothesised that the apparent relation of inverse proportionality between dictionary scale and the average retrieval performance of translations of the test query set was due to an increase in what we termed the *swamping effect* with increasing scale. Dictionaries of larger scale will provide a query translation which will by definition contain more distinct equivalents than one derived using a small-scale dictionary. Not all of these equivalents will be indicative of the content of the original source-language query. In fact, some of them will match irrelevant documents in the collection, resulting in the retrieved document list being "swamped" by many spuriously matching irrelevant documents. The greater the number of equivalents in the query translation, the more such matches will occur and the more performance will be affected or "swamped", thus reducing precision for all precision metrics. R-Prec, in particular, is badly affected by the swamping effect, given that it concentrates on the top M documents only. Recall is affected if this swamping pushes relevant documents below the 1000th document mark in the retrieved document list. We expect the swamping effect's negative influence on precision to counter any positive effects on recall resulting from having a wider range of equivalents to choose from in the larger scale dictionary.

To investigate this, we applied our Difference Runs technique to translations obtained using both our print-derived and automatically-derived dictionaries. If this hypothesis is to be confirmed, one would need to observe drops in performance for a smaller scale dictionary's translation when equivalents from a larger-scale dictionary's translation of the same query were added.

4.5.1 Significance of Hypothesis 4C

We have made the assumption that our smaller-scale dictionaries' entries are more or less subsets of the corresponding entries in *LargeNoRep*. We know this to be the case for our automatically-derived dictionaries and for most of the entries in our print-derived entries. Larger scale means extra equivalents, and so we want to show that it is the lack of these extra equivalents in our smaller-scale dictionaries' entries and no other characteristic of these dictionaries, such as coverage, or the presence of equivalents not given in *LargeNoRep*, that is responsible for the improved performance observed above. Therefore, verifying this hypothesis would reinforce the conclusions reached in Hypothesis 4B above.

4.5.2 Automatically-Derived Dictionaries

First, we compared the translations produced using *AutoVerySmall* with those obtained from *LargeNoRep* and *AutoMediumNoRep*. We selected 8 queries at random from the set of queries for which for which the *AutoVerySmall* translation obtained better R-Prec performance than any other dictionary's translation with R-weighting both enabled and disabled - queries 13, 20, 35, 38, 48, 50, 62 and 68. Four of these *AutoVerySmall* translations were compared with the corresponding *LargeNoRep* translations in a set of Difference Runs, and the other four compared with their *AutoMediumNoRep* translations in a similar manner. A detailed presentation of the results is available in Appendix 4.(ii).

In the set of queries we examined, we found that a large number of the equivalents present in the larger-scale dictionaries' translation but not in the corresponding *AutoVerySmall* translation gave rise to a drop in R-Prec performance when they were added to the latter. For example, in queries 13, 35 and 38 the addition to the *AutoVerySmall* translation of most of the equivalents present in the larger-scale dictionary's translation but not in the former caused R-Prec performance to drop. In other cases, there were some equivalents present in the larger dictionary's translation which on being added to the *AutoVerySmall* translation resulted in an improvement in R-Prec performance, but not enough to counter the negative effect on performance of many others, indicating that the positive effect of having a wider variety of equivalents available was countered by the swamping effect. For example, the addition of *rame* to the *AutoVerySmall* translation of query 48 caused R-Prec performance to improve, but was countered by the negative effect of so many other equivalents in the *LargeNoRep* translation that the overall performance of the *LargeNoRep* translation was less than that of the *AutoVerySmall* translation. These results confirm our hypothesis.

4.5.3 Print-Derived Dictionaries

For the print-derived dictionaries, we selected 9 queries at random from the set of queries whose *Teensy* translation obtained better retrieval R-Prec performance than any other dictionary's translation of the same query with R-weighting both enabled and disabled - queries 4, 8, 13, 16, 21, 35, 49, 53, and 63. We then performed Difference Runs to compare the *Teensy* translations of three of these queries with their *LargeNoRep* counterparts, three others with the corresponding *SGemNoRep* translations, and the last three with the associated *VerySmall* translations. Detailed results of the runs are provided in Appendix 4.(iii). Results were similar to those observed for our automatically-derived dictionaries above.

For example, when we added equivalents present in the *LargeNoRep* translation to the *Teensy* translation of queries 16, 35 and 53, R-Prec performance dropped in most cases. For other queries, there were equivalents in the *SGemNoRep* or *VerySmall* translations which on addition to the *Teensy* translation gave rise to an improvement in R-Prec performance, such as, for example, *chimique* in query 49 and *battre* in query 21. However, other equivalents present in these same translations caused performance to drop when added, resulting in a lower overall score for the larger scale dictionary's translation. Hence, for these queries, the profusion of equivalents in the larger dictionary's translations was responsible for these translations' inferior R-Prec performance. Our hypothesis was confirmed again.

There were, however a minority of queries for which a larger-scale dictionary's translation fared best despite a considerable swamping effect. The led us to pose the following hypothesis:

4.6 Hypothesis 4D: The *Crucial Equivalent Effect* is Responsible for Some Query Translations Bucking the Above Trend

We hypothesised that some smaller-scale dictionaries' translations performed less well than those obtained from a larger-scale dictionary because one or two *crucial* equivalents were missing from the smaller-scale dictionary's translation. We called this phenomenon the *crucial equivalent effect*. Verifying this would reconfirm our findings that query translations are highly sensitive to minor variations in composition, discussed in section 4.3 above. We sought to confirm this hypothesis using our Difference Runs technique, comparing the *LargeNoRep* translations of a set of queries for which a larger scale dictionary obtained the best performance compared with the corresponding smallest-scale dictionary's translations. We would need to find in each case that the observed R-Prec performance difference was due to the absence of a single equivalent, or at most two or three, from the smallest-scale query translations, despite observing a noticeable swamping effect for the relevant *LargeNoRep* translations. Once more, parallel experiments were performed using our print-derived and automatically-derived dictionary sets.

4.6.1 Significance of Hypothesis 4D

If query translations are as sensitive to minor variations in content as we claim, this means that for a given query, we cannot guarantee that using a smaller-scale dictionary will result in improved performance compared to a larger-scale dictionary in every case, leading us to consider ways around this problem. If a smaller-scale and larger-scale dictionary's translations have to be radically different in order for the scale effects on performance discussed above to be inverted, then, a different approach would need to be taken. This hypothesis was therefore important in deciding what we did next.

4.6.2 Automatically-Derived Dictionaries

With R-weighting disabled, there were 13 queries out of the 80 in the test query set for which the *LargeNoRep* translation obtained the highest R-Prec performance, compared to 20 where *AutoMedium-NoRep* fared best and 37 for which the *AutoVerySmall* translation recorded the highest R-Prec score. The relevant statistics with R-weighting enabled were 15, 26 and 47 queries respectively. We selected at random a subset of three queries from the set of queries whose *LargeNoRep* translation performed best with R-weighting disabled - queries 34, 41 and 74 - and three queries for whom similar results were recorded when R-weighting was enabled - queries 12, 14 and 41. Difference Runs were performed to compare the *AutoVerySmall* translations of these queries with their *LargeNoRep* counterparts.

We note that the sample size here is smaller than that of the samples than those selected to confirm previous hypotheses, as there are only a few queries for which the *LargeNoRep* translation obtained the top retrieval performance score. The full details of these difference runs are to be found in Appendix 4.(iv).

Our hypothesis was confirmed. For example, for query 34, the inclusion of the equivalent *construire* in the *LargeNoRep* translation caused R-Prec to jump from 0% to 25%, resulting in an overall improvement in performance between the *AutoVerySmall* translation and the *LargeNoRep* translation despite a concurrent swamping effect. For query 41, the presence of the swamping equivalents *position* and *armee* in the *LargeNoRep* translation lowered the R-Prec score, but the inclusion of *obstacle* or *statut* in the *AutoVerySmall* translation resulted in an improvement. Nearly all equivalents present in the *LargeNoRep* translation of query 12 but not in the corresponding *AutoVerySmall* translation caused R-Prec to drop when they were included in the *AutoVerySmall* translation. However, the equivalents *biologique* and *engrais* gave rise to improvements in performance when added, resulting in an overall improvement in R-Prec performance for the *LargeNoRep* translation compared to the corresponding *AutoVerySmall* translation.

4.6.3 Print-Derived Dictionaries

The *Teensy* translations obtained the best or joint best R-Prec score for 27 of our 80 queries with R-weighting disabled, and for 39 with it enabled. This means that there were quite a few queries for which a translation obtained using one of the other, larger-scale, dictionaries obtained the highest R-Prec score. We randomly selected three queries from the set of queries whose the *LargeNoRep* translation obtained the

best R-Prec performance compared to *Teensy*'s translation with R-weighting both enabled and disabled - queries 3, 15 and 74. (We decided to keep the sample size the same as for the automatically-derived dictionaries above). We then performed Difference Runs to compare the *LargeNoRep* and the *Teensy* translations of these queries. The full details of these runs are available in Appendix 4.(v).

Our hypothesis was confirmed once again. For example, for query 51, the addition of *seisme* to the *Teensy* translation resulted in an improvement in R-Prec performance, as did the inclusion of *importance* and *suite* for query 31, the addition of *voiture* for query 10 and of *exporter* for query 59, despite the presence of a concurrent swamping effect in all of these queries' *LargeNoRep* translations.

4.6.4 Concluding Remarks on Hypothesis 4D

This would suggest that the best way to proceed with query translation would be to combine several dictionaries in the lookup step to ensure that any crucial equivalents were included in the query translation, hopefully without as large a swamping effect as when translating queries with *LargeNoRep* alone.

4.7 Hypothesis 4E: Combining Dictionaries Works Best for Query Translation

Although the majority of queries in our test set benefited from being translated using a small-scale dictionary, we noted above that there were exceptions to this rule. We decided to combine several dictionaries together in translation to try and reduce this crucial equivalent effect while offsetting the resulting increased swamping effect. To combine several dictionaries when translating a given term, we obtained a list of equivalents for that term from each dictionary under consideration and added each of these equivalent lists to the query translation. Equivalents which were provided by more than one dictionary would be repeated in the query translation, resulting in an effective doubling or tripling of their total S-Weight in our retrieval engine's query term weighting scheme, which was set to 1.0 by default. Repeated equivalents would be those supplied by more than one dictionary - the assumption was that an equivalent which was listed by more than one dictionary for a given term was a more important and more mainstream translation than one that appeared in a single dictionary's equivalent list only. These translations would have their weight boosted at retrieval time by virtue of being repeated in the query translation - our retrieval engine makes no attempt to correlate identically spelling query terms, it assumes all terms are independent, an incorrect but commonly mad assumption in IR - hopefully alleviating some of the increased swamping effect we expected to observe. At the same time, we hoped that the crucial equivalent effect would be alleviated, thus resulting in the best of both worlds.

4.7.1 Creating Sets of Combined Translations

To verify this hypothesis, we created four sets of query translations using this combined translation method. Two of these consisted of translations using the print-derived and automatically-derived dictionaries respectively (the *CombinedPrint* and *CombinedAuto* translations), and two others consisted of these combined translations *with all equivalent repetition occurring due to equivalents appearing more than once in a single term's equivalent list removed* (the *CombinedNoRepPrint* and *CombinedNoRepAuto* translations). The swamping effect
would therefore not be alleviated by the repetition of more important equivalents in this latter set of query translations. We expected to observe better performance for the majority of queries for the *CombinedAuto* and *CombinedPrint* translations than for the *CombinedNoRepAuto* and *CombinedNoRepPrint* translations.

4.7.2 Combined Translation Results

The results of these runs are displayed in Figure 4.6. We can see that the swamping effect associated with the *CombinedNoRepPrint* and *CombinedNoRepAuto* matched that affecting the results of *LargeNoRep*. In addition, the differences in performance measured using both AvP and R-Prec, in both the Unweighted and Weighted cases, were significant when the *CombinedAuto* translations were compared with the *CombinedAutoNoRep* translations, and similarly for the translations obtained from the combination

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
LargeNoRep	5	13	6	12	5	15
AutoVS	12	25	11	21	13	26
Teensy	11	24	11	19	12	25
CombPrint	11	23	12	19	13	25
CombAuto	10	21	11	18	12	23
CombNRPrint	5	13	5	12	6	14
CombNRAuto	5	13	5	12	5	15

Figure 4.6: Combining Multiple Dictionaries in Query Translation

Run	UnWRP	UnWAvP	WRP	WAvP
CombAuto v. *CombAutoNoRep*	0.0	0.0	0.0	0.0
CombPrint v. *CombPrintNoRep*	0.0	0.0	0.0	0.0
CombPrint v. *Teensy*	0.575	0.47	0.42	0.575
CombAuto v. *Teensy*	1.0	0.171	0.96	0.33

Figure 4.7: Combined Runs, Significance Tests - Probability of Null Hypothesis

of print-derived dictionaries. This is no surprise as the combined dictionary consisting of all of the equivalents in either set of dictionaries would be of similar scale to *LargeNoRep*.

A query-by-query examination of the R-Prec results demonstrated that this conclusion held for the vast majority of queries. There were three queries for which the *CombinedNoRepAuto* translation had R-Prec performance superior to that of the corresponding *CombinedAuto* translation with R-weighting disabled, and four such queries where R-weighting was enabled. Results for the print-derived dictionary combinations were similar. There were three queries for which the *CombinedNoRepPrint* translations obtained better R-Prec performance scores than the associated *CombinedPrint* translations with R-weighting disabled, and five with R-weighting enabled. (R-Prec was examined with a view to performing some Difference Runs - see below).

These results show that if one wishes to combine dictionaries, it is better not to remove any repeated equivalents - in all cases, the intact combined translations outperformed the corresponding combined translations with repetition removed, and this difference was significant for all metrics tested.

However, these results do not demonstrate that combining dictionaries in translation improves on the performance of translations obtained using a single dictionary. In particular, the results recorded for the *Teensy* and *AutoVerySmall* translations are roughly the same as those reported for the *CombinedPrint* run. We can also see that minor differences in performance between the *Teensy* translations and our sets of combined translations with repetition remaining were not significant. Therefore, our hypothesis, that a combination of dictionaries is better than using a single small-scale dictionary, has **not** been verified.

On examining our results more closely, we noticed that sometimes, repeating an equivalent in a single term's equivalent list in a query translation improved retrieval performance, and that sometimes, it did not. We hypothesised that only *less ambiguous* equivalents would be of benefit to retrieval performance.

4.8 Hypothesis 4F : Repeating Less Ambiguous Equivalents Within a Single Term's Equivalent List Helps Performance, Otherwise, Equivalent Repetition is not Useful

To verify this hypothesis, we used our difference runs technique to compare the performance of the smallest-scale dictionaries with that of the combined runs.

Five queries were selected at random from the set of queries whose *CombinedAuto* translation performed better than its associated *AutoVerySmall*
translation with R-weighting disabled - queries 4, 6, 9, 41 and 55 - and 5 queries for which similar results were recorded with R-weighting enabled - queries 11, 14, 43, 52 and 71. The *AutoVerySmall* translations of these queries were then compared with the corresponding *CombinedAuto* translation in a set of Difference Runs. The full details of these runs are to be found in Appendix 4.(vi).

A swamping effect was noted in most cases, for example, for query 6, most of the equivalents which were present in its *CombinedAuto* translation but not its *AutoVerySmall* translation caused R-Prec performance to drop when they were added to the *AutoVerySmall* translation. However, many repetitions of equivalents within a single term's equivalent list also caused R-Prec performance to drop, not just in query 6, but also in most of the others.

A smaller number of equivalents resulted in an improvement in R-Prec performance when they were repeated in the query translation. For example, for query 4 the repetition of *dechet* and *detritus* caused R-Prec to improve, as did the repetition of the equivalents *desertification* and *bois* in query 9. The repetition of *terrorisme* in query 14 had a similar effect.

Similar results to the above were observed for queries whose *CombinedAuto* translation performed less well than the corresponding *AutoVerySmall* translation. We selected 5 queries at random from the set of queries whose combined translation performed less well than its *AutoVerySmall* translation with R-weighting disabled - queries 12, 13, 19, 63 and 67 - and 5 queries from the set of queries for which similar results were observed with R-weighting enabled - queries 16, 38, 56, 57 and 69. A similar set of difference runs were performed to compare the *CombinedAuto* and *AutoVerySmall* translations of these queries. The details of these runs are presented in Appendix 4.(vii). Results were similar to those discussed above.

Figures 4.8, 4.9, 4.10 and 4.11 display the equivalents which helped and harmed retrieval respectively in these Difference Runs reported in this section only. We define the *degree of ambiguity* of an equivalent as being the number of distinct translations listed for it in the French-English (thus target- to source-language) portion of the Collins Robert Unabridged dictionary. As there was no electronic version of this part of the dictionary available to us, this operation had to be carried out by hand. We emphasise irrespective of the experiment, we *always* use *LargeNoRep* to calculate the degree of ambiguity. This is to ensure consistency of definition across experiments.

We can see that the average degree of ambiguity of the equivalents which helped retrieval performance was considerably lower than that of the others. This can be held to indicate a tendency for less ambiguous equivalents to be more helpful to retrieval on their repetition than those with a higher degree of ambiguity. However, we lack a formally defined criteria for deciding whether or not a given equivalent should be repeated. Some equivalents with a high degree of ambiguity, such as *filer* and *diminuer*, appear in figure 4.8, whereas some which are not ambiguous at all (degree of ambiguity of 1) appear in the other table, for example, *description* and *pollution*.

Therefore, we cannot at this time either verify or reject hypothesis 4F. Further work on this subject is presented in chapter 5.

4.9 Conclusions

Here, we presented our work on dictionary scale. We presented a number of hypotheses based on some control runs, and our main experiments concerning scale, and proceeded to verify or reject them by using the analysis technique of Difference Runs presented in chapter 3.

We reached the following conclusions:

Helped			
Equivalent	**Deg. Ambig.**	**Equivalent**	**Deg. Ambig.**
achever	6	action	8
aller	13	apporter	4
artificiel	8	avortement	2
biologique	3	bois	6
chomage	2	combattre	3
consequence	11	considerer	5
consommation	7	dauphin	3
debris	11	dechet	11
desertification	3	detritus	3
dette	1	diminuer	16
diminution	8	diriger	21
evaluer	5	exploitation	5
fabrication	6	filer	13
fixer	11	france	1
grischun	1	grossesse	1
international	4	interruption	3
kidnapping	2	lutter	3
mer	2	obstacle	3
ocean	2	ordure	7
origine	5	oua	1
pecher	8	pologne	1
position	5	production	10
rumantsch	1	statistique	2
statut	2	automobile	9
taux	2	terrorisme	1
tuberculose	1	vin	1
volontaire	7		
Average	**5.0**		

Figure 4.8: Equivalents Which Helped Retrieval in Combined Translations

Harmed			
Equivalent	**Deg. Ambig.**	**Equivalent**	**Deg. Ambig.**
accord	6	aerer	4
affaire	15	africain	1
air	11	aller	13
armee	2	artificiel	8
assiduite	3	assurer	10
attitude	4	auto	3
avancer	16	but	8
calculer	7	campagne	4
chamois	4	combat	3
combattre	3	commission	5
connaitre	11	contamination	2
continental	2	conversion	4
deboucher	15	decrire	2
depeindre	1	description	1
devenir	3	difficulte	3
direction	25	diriger	21
disposition	17	donner	16
echelonner	8	economique	3
effectuer	8	entrainer	12
europeen	2	eventualite	3
fabrication	6	fond	14
garantir	4	gerer	2
grand	24	harmonie	3
hausse	2	illegal	3
impact	2	impliquer	3
industrialiser	2	industrie	4
lancer	30	langage	1
continued...			

Figure 4.9: Equivalents Which Harmed Retrieval in Combined Translations - Part 1

Harmed (con.)			
Equivalent	**Deg. Ambig.**	**Equivalent**	**Deg. Ambig.**
large	7	legitime	8
lever	19	loin	3
loyer	1	lutter	3
marche	20	miltaire	3
monde	5	milieu	12
national	2	obstacle	3
organique	1	organisation	6
oriental	3	paix	5
patrie	3	pays	3
peau	5	peril	2
place	12	planche	11
politique	7	pollution	1
pose	17	poser	30
position	5	proceder	7
procedure	1	processus	2
profanation	6	quatrieme	6
raison	5	raisonner	4
rapporter	16	rayon	11
reapparition	1	redemarrage	2
reduire	17	relever	37
rentable	1	ressortissant	1
roche	1	savoir	4
siecle	3	situation	5
soutenir	11	subir	10
suisse	3	survie	2
Continued...			

Figure 4.10: Equivalents Which Harmed Retrieval in Combined Translations - Part 2

Harmed (con.)			
Equivalent	**Deg. Ambig.**	**Equivalent**	**Deg. Ambig.**
synthetique	1	taille	14
trafic	6	traiter	10
tranquillite	5	transformation	5
transformer	7	vendre	2
vers	6	vitesse	5
zele	1		
Average	**7.0**		

Figure 4.11: Equivalents Which Harmed Retrieval in Combined Translations - Part 3

- The upper bound on AvP and R-Prec is around 35% using our system.
- The retrieval performance of query translations is very sensitive to minor variations in content, and therefore sensitive to gaps in a translation system's knowledge or to translation errors.
- Using smaller-scale dictionaries to translate queries results in better retrieval performance, provided 100% coverage is assured using our coverage compensation process. There appears to be a relationship of rough inverse proportionality between dictionary scale as we measure it and retrieval performance.
- This rough inverse proportionality is due to the *swamping effect* increasing with dictionary scale.
- Some queries buck this trend because the smaller-scale translations omit one or two *crucial equivalents* which are so important for retrieval that they counter the swamping effect for translations obtained from larger-scale dictionaries. This again demonstrates query translations' sensitivity to gaps in translation knowledge.
- Combining several dictionaries of differing scale did not appear to benefit retrieval performance, because the repetition introduced by dictionary combination is not universally beneficial, and not enough to counter the substantial swamping effect.
- It appears that only those equivalents with a low *degree of ambiguity* benefit retrieval performance by their repetition in a single term's equivalent list in the query translation.

This last finding was promising, but no well-defined criteria of what constitutes "more ambiguous" and "less ambiguous" have been defined. The next chapter will explore the effects of equivalent repetition on query translation performance more thoroughly, as well as looking alternative dictionary combination strategies and at the effects of varying small-scale dictionary coverage rate prior to the application of coverage compensation using *LargeNoRep*.

Chapter 5

Differences Between Similar Dictionaries, Coverage, Equivalent Repetition and Retrieval Performance

In chapter 2, we explained that we were interested in investigating the effect of dictionary characteristics on the retrieval performance of associated query translations. The last chapter dealt with our experiments on dictionary *scale* and the *crucial equivalent effect*, and described some preliminary experiments on the effect of *equivalent repetition within a single term's equivalent list* in query translations on retrieval performance using our retrieval engine.

This chapter builds on our work on the crucial equivalent effect. We present the notion of *CLIR dictionary source*, examining the effects on retrieval performance of minor variations in entry content between dictionaries of similar scale derived from similar sources. We also propose an alternative method of dictionary combination aimed at reducing the observed crucial equivalent effect.

Following on from this, we discuss some more in-depth experiments on the subject of equivalent repetition, leading us to conclude that more ambiguous equivalents should have their weight adjusted in the query translation so as to be less important than less ambiguous ones. We also return to the issue of coverage, this time within the context of our coverage compensation procedure, first discussed in chapter 3.

As in the previous chapter, the work described here is presented as a series of hypotheses, which are verified or disproved in turn.

5.1 The Effect of Minor Variations in CLIR Dictionary Entry Content on the Retrieval Performance of Query Translations

In chapter 3, we showed how a CLIR dictionary could be obtained from a conventional printed bilingual dictionary. Following this, we presented a number of CLIR dictionaries which we derived using this procedure from printed bilingual dictionaries aimed at language learners. However, printed bilingual dictionaries aimed at language learners constitute a very small part of the universe of potential CLIR dictionary sources available to us. By *source* we mean a linguistic resource, or entity containing a linguistic resource, which may be processed in some way (not necessarily using the procedure mentioned above) to obtain a CLIR dictionary.

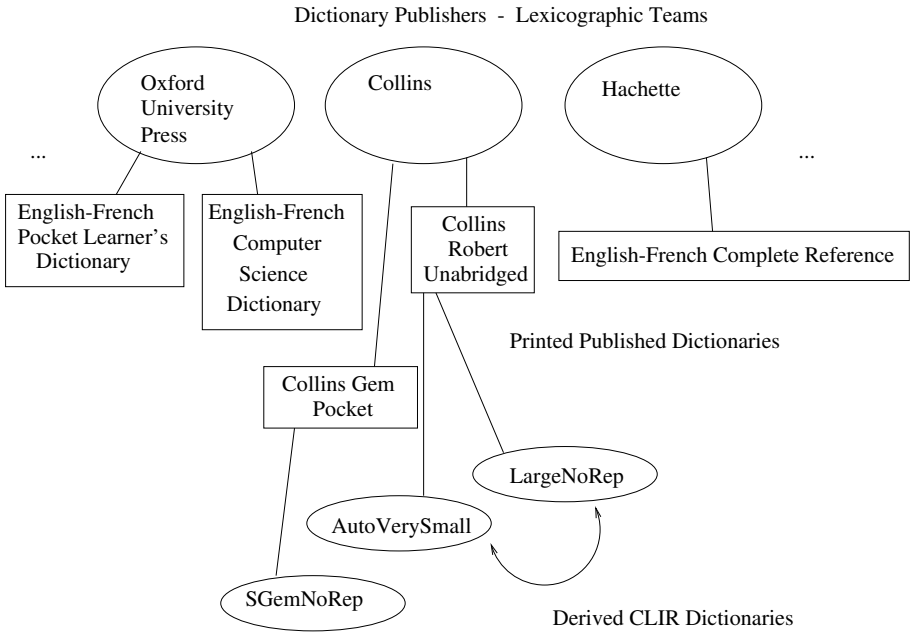

Figure 5.1: Part of the CLIR Dictionary Source Universe

5.1.1 The CLIR Dictionary Source Universe

In fact, there is no reason why we have to use a printed dictionary at all to derive a CLIR dictionary - all we need is some sort of resource from which we can obtain a set of entries consisting of headwords paired with equivalent lists. We could extract this information from, say, a modified list of words from a spell checking application, a list of words downloaded from the Internet or an electronic bilingual dictionary file.

Printed bilingual dictionaries are not themselves identical in content. Apart from the obvious issues of scale, one can imagine, for example, a reference dictionary, such as a bilingual version of the Complete Oxford English Dictionary, where equivalents are given in chronological order, and not frequency of use as is typically the case with language learner's dictionaries. Also, printed dictionaries with a restricted domain, such as technical, legal or tourist-oriented dictionaries, do exist. One would expect the level of coverage afforded by and the subject area covered by these different types of printed dictionary to vary greatly, as they are aimed at distinct markets with varying needs. A CLIR dictionary could potentially be derived from any of the above. In addition, a number of CLIR dictionaries of differing scale may be obtained from a single printed dictionary.

Figure 5.1 shows a part of what we call the *CLIR Dictionary Source Universe*. It shows how the lexicographic teams of various dictionary publishing firms issue multiple dictionaries aimed at different markets, which can include several directed at language learners, the subset of this Universe in which we are interested. The Figure depicts how three of our own CLIR dictionaries, *LargeNoRep, SGemNoRep* and *AutoVerySmall* fit into this subset of the Source Universe. Our depiction of the CLIR Dictionary Source Universe here is limited to the part of the Universe inhabited by and related to printed dictionaries, as this constitutes the area of interest of our research. We concentrated our work on a very small part of the potential CLIR Dictionary Source Universe, CLIR dictionaries derived from published printed dictionaries aimed at language learners, because nearly all of the dictionary-based CLIR systems reported in the literature employed CLIR dictionaries obtained from such a source.

5.1.2 Choosing Dictionaries Derived from Similar Sources

We assessed the impact of *small differences in entry content* between three CLIR dictionaries derived from three pocket-sized printed bilingual dictionaries aimed at language learners issued by different publishers. One would expect these three CLIR dictionaries' printed sources to be similar in content, even though

they were issued by different publishers, as the market addressed and the level of detail provided in the printed dictionary entries were similar. Therefore, we expected the differences in entry content between these CLIR dictionaries to be slight, and it is the effect on retrieval performance of slight rather than major differences in which we were interested.

In chapter 4, some control experiments revealed that minor differences in content between pairs of query translations could have a considerable effect on their relative retrieval performance. Would we observe a similar effect for query translations obtained using our three dictionaries in our new experiments? The dictionaries employed in these new experiments were of similar scale, in order to avoid scale affecting in our results. (We note that scale was calculated using the 385 terms in our test query set only, as described in chapter 3).

5.1.3 Presenting Our Three Similar Dictionaries

We used the dictionary *SGemNoRep*, discussed in chapters 3 and 4, and derived two further dictionaries, *InsightNoRep* and *SLangNoRep*, using two printed bilingual dictionaries, the Insight Tourist Dictionary and the Langenscheidt Universal Dictionary respectively, by following the method described in chapter 3. These printed dictionaries were, like the Collins Gem dictionary from which *SGemNoRep* was obtained, both pocket-sized English-French dictionaries aimed at GCSE-level language students. (Despite the name, the Insight Tourist Dictionary is a conventional language learner's pocket dictionary and not a tourist phrase book). Repeated equivalents in *InsightNoRep* and *SLangNoRep* entries were removed as for *SGemNoRep* (see in chapter 3 for a an account of this procedure). Scale of 2.4, 2.2 and and 2.1 was recorded for *SGemNoRep, SLangNoRep* and *InsightNoRep* respectively.

71 source-language terms in our test query set did not have entries in *InsightNoRep*, whereas 70 terms were missing from *SLangNoRep* and 59 from *SGemNoRep* (prior to coverage compensation). Therefore, the coverage compensation process described in chapter 3 and already applied to *SGemNoRep*, involving the insertion of entries from *LargeNoRep* into a smaller-scale dictionary for any terms missing from the latter, was also applied to *InsightNoRep* and *SLangNoRep*. We can now be assured of a coverage rate of 100% for all our experiments, thereby ensuring that lack of coverage did not introduce any artifacts into our results. In addition, nearly two-thirds of the terms originally missing from each dictionary were also missing from at least one other - indicating that a similar range of terms was covered by each dictionary. This satisfies the criteria for our first set of experiments, that we are using dictionaries which differ slightly, and not greatly, from one another in content. We were ready to pose the first hypothesis of this chapter:

5.1.4 Hypothesis 5A: Retrieval Performance will be Different for Each Dictionary's Translations Because of the Sensitivity of Retrieval Performance to Minor Variations in Query Translation Composition

To verify this hypothesis, we would need to observe significant differences in performance between the three sets of translations of the test queries obtained using these three dictionaries. In addition, analysis of these results should reflect the conclusions reached in chapter 4 regarding the sensitivity of retrieval performance to minor variations in query translation content: that observed major differences in performance between two translations of the same query would need to be due to at most two or three points of difference between them. Retrieval performance was compared using AvP, R-Prec and (sometimes) DC-20 for the entire test query set and also on a query-by-query basis.

5.1.5 Significance of Hypothesis 5A

Verifying this hypothesis would reinforce our earlier conclusions that that small gaps in the translation knowledge embodied in a particular dictionary could result in lower performance for an associated query translation than if it was translated using some other, very similar dictionary. Therefore, the micro-content of individual dictionary entries does matter. Since we have no way of knowing which queries will suffer as a result of such "knowledge gaps", this makes choosing a suitable dictionary for a CLIR system extremely difficult, even if we have an idea of the scale of dictionary desired. This provides support for

Run	**UnWAvP**	**WAvP**	**UnWDC20**	**WDC20**	**UnWRP**	**WRP**
SGemNoRep	9	20	10	19	10	22
SLangNoRep	8	21	10	18	10	23
InsightNoRep	9	21	11	19	10	23

Figure 5.2: Add-All-Equivalents Runs for Three Dictionaries

the theory that an effective dictionary-based CLIR system would draw from a wide range of potential dictionary sources.

5.1.6 Comparison of Add-All-Equivalents Translations for Our Three Dictionaries

We translated the test query set of 80 English queries using each of these three coverage-compensated dictionaries in turn using the Add-All-Equivalents method as for our dictionary scale experiments reported in chapter 4. The results of these runs are displayed in Figure 5.2, showing AvP, R-Prec and DC20 for both Weighted and Unweighted runs (see chapter 3). Once more, our results are expressed as percentages rounded to two figures, rather than as numbers between 0 and 1, and the percent sign implied.

Initially, our hypothesis appears to have been rejected, as results with retrieval engine R-weighting both enabled and disabled were similar for all three sets of translations, for all three evaluation metrics. However, a query-by-query examination of the results revealed that there was considerable variation in performance between the translations of individual queries, but that the effects of all these variations happened to cancel each other out overall for our test query set. One cannot guarantee that a cancellation effect would be observed for every possible set of queries, unless the test query set was very large. Therefore, it is certainly possible that for an arbitrary-sized set of queries, a substantial overall difference in performance could be observed depending on the choice of dictionary employed to translate them. We concluded that for a given query, there was a considerable likelihood of different results being recorded depending on the dictionary employed.

Figure 5.3 displays a fragment of the query-by-query R-Prec results for each dictionary's translation with R-weighting enabled. We can see that, for example, for query 13, *SGemNoRep*'s translation obtained R-Prec of 6%, *SLangNoRep*'s translation 23% and *InsightNoRep*'s translation 20%. If our query set were to consist only of queries 13, 14, 16 and 17, overall R-Prec would be 26%, 29% and 30% for the *SGemNoRep*, *SLangNoRep* and *InsightNoRep* translations of these queries respectively.

This verifies the first part of hypothesis 5A: that significant differences in retrieval performance were observed between different translations of the same query, even though the dictionaries employed to translate the queries were very similar in source and scale. We now need to show that these performance differences occurred due to minor and not major differences in content between the different query translations.

5.1.7 Variations in Query Translation Content

We wanted to show that the performance differences observed were due to minor differences in content between the sets of query translations obtained using our three dictionaries. We defined two translations of a given query as varying slightly in content if there were only a small number of *points of difference* (defined in section 3.4) between them. The notion of *points of difference* was defined in the description of our *Difference Runs* technique in chapter 3. A point of difference between two translations of a given query is the presence of a single equivalent in the first which is not present in the latter, or the converse. The idea is similar to that involved in string editing, in that enumerating the points of difference between two translations of a query involves counting the number of single operations (equivalent additions and deletions) necessary in order to transform one of these query translations into the other.

If the majority of the performance differences observed between alternative translations of queries in our test set were indeed found to be due to a small number of points of difference, the second part of our hypothesis will have been verified. (We have not defined precisely what constitutes a "small" number of points of difference at this stage, so this remains an intuitive concept, however, one or two certainly constitutes a "small" number). If we were to find that the query translations for whom differing

QueryNum	*SGemNoRep*	*SLangNoRep*	*InsightNoRep*
08	17	20	22
09	15	15	15
10	10	24	21
11	0	0	0
12	23	03	03
13	06	23	20
14	20	18	18
15	02	0	0
16	33	44	33
17	43	34	49
18	0	0	0
19	17	17	17
20	44	44	44
21	6	3	3
22	0	0	0

Figure 5.3: Query-By-Query R-Prec (fragment of)

performance was recorded were in the main quite different from one another in content, having very many points of difference between them, the second part of our hypothesis will have been rejected.

5.1.8 Difference Runs

Looking at the R-Prec results where retrieval engine R-weighting was enabled, we selected at random three queries from the set of queries for which the *SGemNoRep* translation obtained higher R-Prec performance than the other two translations, queries 48, 52 and 55, and three queries from the set of queries whose *SLangNoRep* translation fared best, queries 10,13 and 59, and three further queries from the set of queries whose the *InsightNoRep* translation recorded the best R-Prec performance score, queries 17, 50 and 58. We looked at the Weighted case only as results were similar for both the Weighted and Unweighted runs. We then performed Difference Runs between each of these translations and the next highest scoring translation for each query. We note again that R-Prec alone was employed in our Difference Runs because looking at all three measures - AvP, R-Prec and DC20 - would have been time-consuming and unwieldy. The sample size here is similar to that employed for some of the Difference Runs in chapter 4, and constitutes roughly a sixth of the set of available queries in each case.

The second part of hypothesis 5A was verified. For example, for query 58, the *InsightNoRep* translation scored R-Prec of 83%, compared to 50% for the corresponding *SGemNoRep* translation. This difference in performance was due to a single point of difference between the two translations, namely the presence of the equivalent *charge* in the *SGemNoRep* translation. Similarly, the *SLangNoRep* translation of query 59 performed better than its *SGemNoRep* counterpart due entirely to the presence of the equivalent *deprimer* in the latter. The omission of the equivalents *vers, moyen* and *orient* from the *SLangNoRep* translation of query 13 lead to it having the best R-Prec performance of the two, as did the omission of the equivalents *juridique* and *statistique* from the *InsightNoRep* translation of query 55. (We shall see the results of not omitting these equivalents in our combined translations below). A full set of results are provided in Appendix 5.(i).

5.1.9 Concluding Remarks on Hypothesis 5A

Hypothesis 5A has been verified. Retrieval performance is sensitive to small differences in content between query translations obtained from dictionaries of similar scale and content, derived from similar sources aimed at the same market. This means that the crucial equivalent effect can have a significant impact on the retrieval performance of a single given query translation and cannot be ignored, even when the range of dictionaries available for incorporation into the CLIR system is highly restricted. On the other hand, using a single small-scale dictionary reduces the swamping effect considerably. This led us to consider an alternative method for combining several similar dictionaries for query translation, to see if this new method could offset the crucial equivalent effect without being negated by the swamping effect.

5.2 Combining All Three Dictionaries - A Solution to the Crucial Equivalent Effect?

The dictionary combination method proposed in chapter 4 involving the combination of several coverage compensated dictionaries of different scale was not successful at reducing the crucial equivalent effect while keeping the swamping effect to a minimum. Here, we look at an alternative way of combining dictionaries where we combine the three small-scale dictionaries discussed in the previous section, again to attempt to reduce the anticipated crucial equivalent effect of using a single small-scale dictionary. We hoped that the reduced swamping effect associated with smaller-scale dictionaries would result in better performing query translations than our previous efforts at dictionary combination.

By combining three dictionaries for query translation, we mean that for each source-language query term, we obtain the full list of equivalents for that term from each dictionary, combine all of these equivalent lists in a single list for that term through list concatenation, and then added this complete list of equivalents to the query translation. There will be an increased swamping effect due to the greater number of distinct equivalents obtained for each query term using this method. There will also be a lessening of the crucial equivalent effect, as we expect the dictionaries to compensate for the others' knowledge gaps by providing equivalents the others do not. We hoped that this lessening of the crucial equivalent effect would more than counter the increased swamping effect.

5.2.1 Two Sets of Combined Translations

We obtained two sets of combined translations. For the first, *Combined*, no attempt was made to remove any repeated equivalents from the complete list of equivalents obtained for each source-language query term using our three small-scale, coverage compensated dictionaries. For the second, *CombinedNoRep*, any second or subsequent occurrence of an equivalent in the complete equivalent list obtained for any source-language query term was removed before the list was added to the query translation. We addressed equivalent repetition in query translations briefly in the last chapter and expected it to have an impact on our results here.(Note that we do not address the possibility of an equivalent being present twice in the query translation due to being a potential translation of more than one query term. We chose to look at equivalent repetition within a single term's equivalent list only, as we did not have enough time to carry out all possible investigations). The *CombinedNoRep* runs allow us to observe the crucial equivalent and swamping effects without any interference from equivalent repetition. The *Combined* runs allow us to consider the effect of repetition in a subsequent section. The results for both sets of runs, with retrieval engine R-weighting both enabled and disabled, are displayed in Figure 5.4.

5.2.2 Retrieval Performance of *CombinedNoRep* Translations

The *CombinedNoRep* runs did not perform significantly better overall than any of the individual dictionaries on their own (see figure 5.4). This means that the reduction of the crucial equivalent effect in these translations was not enough to counter the increased swamping effect occasioned by an increased number of distinct equivalents being provided for each query term. Therefore, combining similar dictionaries is not the solution to the crucial equivalent effect described above - it may be that this latter effect is something the users of a dictionary-based CLIR system simply have to live with.

However, it could be that allowing equivalent repetition is the key - the *Combined* translations performed on average at least as well as any of the individual dictionaries' translations.

5.2.3 The Effect of Equivalent Repetition on Combined Translation Performance

The *Combined* translations obtained a much better average retrieval performance score than the *CombinedNoRep* translations. This indicated that the swamping effect of combining three dictionaries was somehow being offset by the presence of repeated equivalents in the *Combined* translations due equivalents being provided as a translation for a given term by more than one dictionary in our combination. We wanted to find out how this type of equivalent repetition was benefiting retrieval performance.

Runs	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
Combined	10	22	15	27	12	24
CombinedNoRep	8	20	12	25	10	22
SGemNoRep	9	20	10	19	10	22
SLangNoRep	8	21	10	18	10	23
InsightNoRep	9	21	11	19	10	23

Figure 5.4: Results of Running the Combined Query Translations

Runs Compared	UnWRP	UnWAvP	WRP	WAvP
Combined v. *CombinedNoRep*	0.086	0.0	0.007	0.0
Combined v. *SGemNoRep*	0.022	0.159	0.151	0.077
Combined v. *InsightNoRep*	0.195	0.225	0.211	0.181
Combined v. *SLangNoRep*	0.003	0.0	0.049	0.0

Figure 5.5: Significance Tests for Combined Runs - Probability of Null Hypothesis

However, although the *Combined* translations also outperformed all three individual dictionaries' translations on average, significance testing revealed that these differences were not significant (see Figure 5.5). A query-by-query examination also revealed that, whereas in some cases, the relevant *Combined* translation outperformed all three dictionaries' translation, this was not always the case. This means that yet again, our attempts to improve retrieval performance using dictionary combination have failed. Hypothesis 5.B has not been verified.

So, why is equivalent repetition within a single term's equivalent list sometimes beneficial and sometimes not, and what is the role of the swamping effect here? We posed the following hypothesis:

5.3 Hypothesis 5B-1: Repetition of the More Important Equivalents within a Term's Equivalent List is Responsible for the Improved Performance of some of the *Combined* Query Translations

The equivalents which were repeated in a single term's equivalent list in the *Combined* translation of a given query were those which were supplied as a possible translation of a given term by more than one dictionary - thus the final equivalent list for the relevant term contained more than one occurrence of a given equivalent. All three dictionaries were derived from printed dictionaries aimed at language learners, which tend to feature the most common equivalents of a given term, because these are the most important for a language learner to know in order to acquire a core vocabulary in the language being studied. Therefore, we would expect those equivalents which featured in more than one dictionary's entry for a given term to be the more commonly used and therefore more important translations of that term. (This would not necessarily be the case if, for example, our dictionaries had been derived from printed reference dictionaries or a spelling checker). This repetition of important equivalents within terms' equivalent lists counters the swamping effect inherent in dictionary combination by increasing the weight of these important equivalents, thereby decreasing the influence of more unusual equivalents on retrieval results.

To verify this hypothesis, we would need to demonstrate that all such equivalent repetition in the *Combined* query translations contributed to the improved retrieval performance observed with respect to the *CombinedNoRep* translations. We know that not all of the *Combined* translations performed better than their *CombinedNoRep* counterparts - so some of the repetition must have harmed rather than

helped retrieval. In addition, we noted in chapter 4 that not all repetition was beneficial to retrieval performance - only the less ambiguous equivalents, where ambiguity was measured with reference to the number of distinct source-language translations listed in the Collins Robert Unabridged Dictionary for a given equivalent, appeared to benefit retrieval performance by their repetition in the experiments reported there. We expected to observe a similar pattern in our experiments here. Therefore, we modified our hypothesis:

5.3.1 Hypothesis 5B-2: The Repetition of *Less Ambiguous* and therefore *Less Frequent* Important Equivalents Within a Single Term's Equivalent List improves Retrieval Performance

A fairly crude type of probabilistic equivalent S-weighting (combination of T- and Q-weighting) is being applied when equivalents are repeated in a *Combined* translation, where the probability of an equivalent being a good translation of a given term is calculated based on the number of dictionaries in which it appears and an additional T-weight applied accordingly. The default T-weight assigned to an equivalent is 1.0, and this T-weight is multiplied for each repetition of the equivalent in the query translation. Other researchers have attempted similar types of T-weighting, for example, by using a single large-scale dictionary and counting the number of occurrences of an equivalent in the dictionary entry for a given term [44], or by employing bilingual corpora to generate the T-weights according to probabilistic formulae [43].

It may appear that there is no need for additional S-weighting, as existing monolingual IR techniques such as *idf* weighting, implemented by our retrieval engine's R-weighting strategy, take care of frequency related equivalent weighting. (One expects frequency and ambiguity to be related - the more possible meanings an equivalent has in the target language, the more likely one is to encounter it in a random sample of text in that language). However, observations based on a monolingual setting are not always valid in a cross-language setting. There tend to be so many equivalents in the query translation that this *explosion of terms*, as it is called, dilutes the original meaning of the query and overwhelms the retrieval engine R-weighting scheme, irrespective of the relative frequency or ambiguity of the equivalents in the query translation. Therefore, the R-weighting strategies employed might benefit from some modification, in order to take the particular needs of CLIR into account. (As stated in previous chapters, R-weighting refers to the retrieval weight calculated by the retrieval engine itself, whereas S-weighting denotes any additional weights applied as a multiplier to the retrieval engine R-weight, generally passed to the engine in the query translation. We call S-weights applied before translation Q-weights, and those after translation T-weights).

We have already established that, due to the character of the sources from which our three dictionaries were derived, the equivalents which were repeated in terms' equivalent lists in the *Combined* query translations were those which are considered the most common and therefore most important translations of a given term from the point of view of a language learner. To verify our modified hypothesis 5B-2, we need to establish a correlation between the degree of ambiguity of these repeated important equivalents and the effect of their repetition on query translation performance. In addition, we need to confirm that equivalent ambiguity and frequency in the target-language document collection are directly related in our experiments.

In particular, equivalents of a low degree of ambiguity should lead to improvements in performance on being repeated, whereas the converse should be observed for equivalents of a comparatively high degree of ambiguity. This would lead us to conclude that the existing retrieval engine R-weighting strategy could benefit from further ambiguity-related (and therefore frequency-related) S-weighting in a CLIR setting. As we shall be looking at query translations which have been obtained using a combination of small-scale dictionaries, it is worth restating that the degree of ambiguity of a French equivalent is *always* calculated using the French-English portion of the Collins-Robert Unabridged Dictionary, regardless of the dictionaries used for query translation, so that this measure can be consistent across all experiments.

5.3.2 Difference Runs

To verify our hypothesis, we performed a set of Difference Runs to determine the effect of the T-weighting implicit in equivalent repetition by comparing the *Combined* and *CombinedNoRep* translations of a sample

set of queries. Since the only difference between the *Combined* and *CombinedNoRep* translations of a given query is the presence of equivalent repetition in the former, we can observe the effect of equivalent repetition in isolation by comparing pairs of these two sets of query translations. Once more, we look at R-Prec results only.

We selected randomly a set of 4 queries from the set of queries whose *Combined* translation obtained the best retrieval performance of the two with retrieval engine R-weighting disabled - queries 8, 14, 68 and 69, and 4 more from the set of queries where the converse was the case - queries 12, 17, 41 and 50. Two similar sets - queries 3, 4, 16 and 58 and queries 15, 24, 35 and 78 - were selected where retrieval engine R-weighting was enabled. Difference runs were performed between each pair of query translations. The sample size here, 16 queries in total, constitutes 20% of our entire test query set.

With retrieval engine weighting disabled, for both sets of queries, repetitions which helped and harmed R-Prec performance were observed. The only difference between the queries for which the *Combined* translation obtained the best R-Prec performance score and those where the *CombinedNoRep* translation fared best was the extent of the beneficial and harmful effects. In one case the cumulative effects of the beneficial repetitions was enough to counter any negatively affecting repetition or swamping effects, resulting in the *Combined* translation obtaining the best R-Prec score. In the other, it was not, and so the *CombinedNoRep* translations had the highest R-Prec score.

For example, with retrieval engine R-weighting disabled, for query 8, the repetition of the equivalents *augmenter, limite* and *route* boosted R-Prec, while many other repetitions caused R-Prec to drop. Similar mixed results were observed for query 14 (the repetition of *terrorisme* improved R-Prec performance, whereas repeating *international* lowered the R-Prec score). In query 68, the repetition of *homosexual* improved things, whereas similar repetition of other equivalents harmed R-Prec performance. In query 41, the repetition of *allemagne* increased the R-Prec score considerably, whereas the inclusion of extra instances of *difficulte, situation* and *necessiter* caused R-Prec performance to drop. Results were similar with retrieval engine R-weighting enabled. Full details of these runs are to be found in Appendix 5.(ii).

5.3.3 Ambiguity, Repetition and Retrieval Performance - Is there a Correlation?

Figures 5.6, 5.7 and 5.8 display the target-language document collection frequency and degree of ambiguity of each equivalent that helped and harmed retrieval respectively in these runs by being repeated within a single term's equivalent list. There are more equivalents with a lower degree of ambiguity in the list of equivalents whose repetition helped retrieval than in the list of equivalents which harmed retrieval on their repetition in the query translation. Furthermore, the average degree of ambiguity for Figure 5.6 is lower than that displayed in Figures 5.7 and 5.8. However, there are equivalents of comparatively low and high degree of ambiguity in both tables, and some equivalents, such as *agriculture*, occur in both. In addition, the difference in average degree of ambiguity is not enormous, being just under 1.0, and the standard deviation in both cases is quite large. Therefore, this result should be viewed as suggestive only. Larger scale investigations need to be carried out if we wish to verify our hypothesis.

5.3.4 Ambiguity and Frequency

Figures 5.6, 5.7 and 5.8 also show the frequency of each of these translation equivalents in the target language document collection. Contrary to what one would expect, the link between ambiguity and frequency often observed in the monolingual retrieval experiments does not appear to apply in the cross-language case. For example, although the average degree of ambiguity of equivalents listed in Figure 5.6 is less than in Figures 5.7 and 5.8, the converse is true of the collection frequency. In addition, the standard deviation in both cases is quite large, indicating the presence of many outliers. Furthermore, the graphs in Figures 5.9 and 5.10, which plot degree of ambiguity against collection frequency for both the equivalents which helped and harmed retrieval, shows no clear correlation between degree of ambiguity and frequency. Although this may be due to the relatively small sample size, it is potentially of some interest. Further investigation was outside the scope of these experiments. This means that one part of our hypothesis has not been verified.

Equiv	Freq	Degree	Equiv	Freq	Degree
accident	6902	6	adriatique	69	1
agriculture	2707	2	algue	124	2
allemagne	4725	1	arreter	5870	18
combat	3453	3	combattre	1146	3
cote	8578	29	couple	1070	3
cuir	93	5	decliner	104	15
drogue	3936	3	homosexuel	332	1
individu	603	3	industrie	4675	4
international	14492	4	italien	6168	1
militaire	14526	3	ordure	307	7
peluche	27	5	pomme	281	10
porter	8518	37	refuser	6230	10
rejet	1145	21	route	5503	5
slovenie	334	1	terre	3437	8
terrorisme	855	1	tourisme	1789	3
trafic	6031	6	tuberculose	130	1
vitesse	1306	5			
Average	**3498**	**6.9**	**StdDev**	**3979**	**8.2**

Figure 5.6: Equivalents Which Helped Retrieval Performance, Their Collection Frequency and Their Degree of Ambiguity

Equiv	Freq	Degree	Equiv	Freq	Degree
agriculture	2707	2	article	3435	6
augmenter	6080	10	avancer	1662	16
baisser	2877	26	barriere	445	4
chemin	2035	5	circulation	3107	5
commerce	5394	10	commercialiser	240	1
concentration	853	1	conduire	3302	12
consequence	2608	11	considerable	827	6
considerablement	503	3	contenir	1887	12
cote	8578	29	declin	163	5
decliner	104	15	difficulte	2337	3
echanger	997	3	endiguer	99	5
entrainer	2340	13	gagner	2429	7
gain	843	9	grand	24550	24
gros	2181	17	impact	690	1
importer	1624	3	information	8223	4
international	14493	4	large	2577	7
limiter	2760	4	lourd	2081	21
medicament	1412	2	mesure	13281	11
mesurer	541	12	metier	654	10
monde	7428	5	mondial	5466	2
monnaie	1573	3	necessiter	759	4
organique	136	1	ours	175	2
patrie	341	3	pied	2435	13
plage	345	6	police	21399	8
popularite	195	1	porter	8518	37
possibilite	3308	1	principal	8446	10
Continued...					

Figure 5.7: Equivalents Which Harmed Retrieval Performance, Their Collection Frequency and Their Degree of Ambiguity - Part 1

Equiv	Freq	Degree	Equiv	Freq	Degree
possibilite	3308	1	principal	8446	10
raisonner	23	4	rapidite	122	7
reapparition	61	1	recherche	5332	16
redemarrage	61	2	referendum	1849	1
renseignement	1391	4	rentable	198	1
rouler	463	19	situation	9572	5
souci	445	3	supporter	662	9
tige	10	12	vaste	1060	7
vendre	2837	2	violent	2343	5
voie	4476	6			
Average	**3222**	**7.8**	**StdDev**	**4604**	**7.2**

Figure 5.8: Equivalents Which Harmed Retrieval Performance, Their Collection Frequency and Their Degree of Ambiguity - Part 2

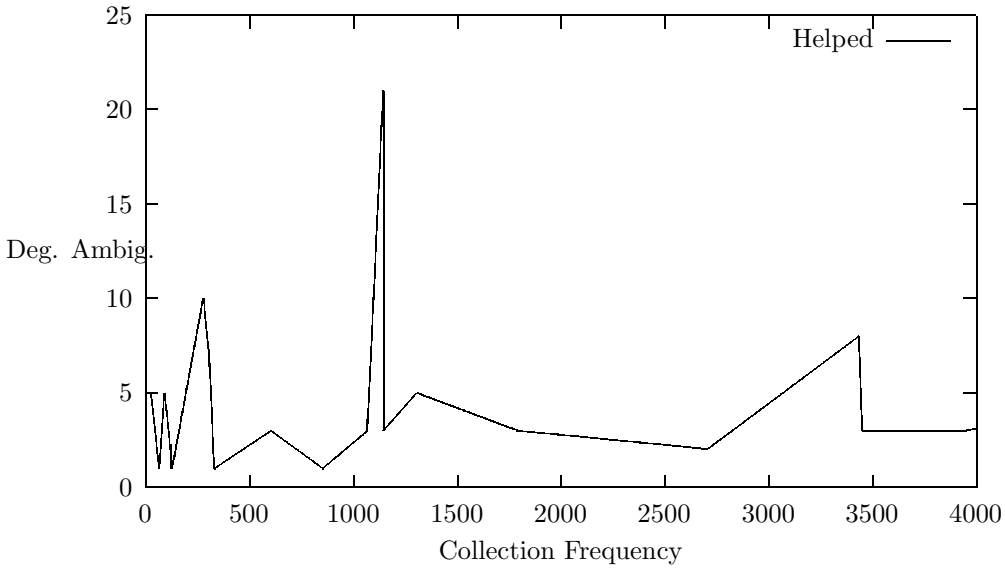

Figure 5.9: Equivalents Which Helped Retrieval Performance, Collection Frequency v. Degree of Ambiguity

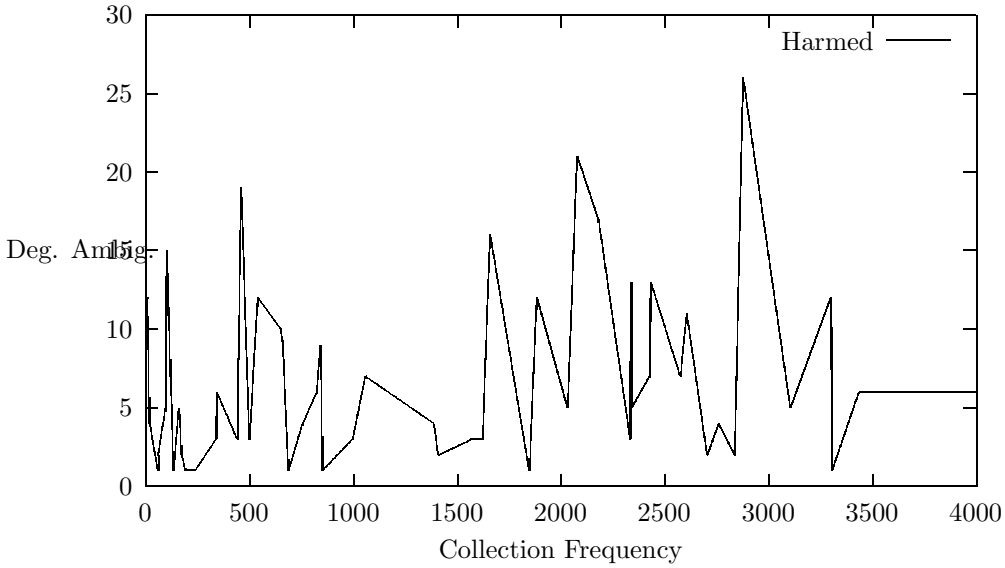

Figure 5.10: Equivalents Which Harmed Retrieval Performance, Collection Frequency v. Degree of Ambiguity

5.3.5 Concluding Remarks on Hypothesis 5B-2

Some parts of our hypothesis have been verified - less ambiguous equivalents do seem to warrant an extra S-weight in query translations in addition to the R-weight assigned by the retrieval engine. This does not appear to be linked to low frequency of these equivalents in the target-language document collection. However, these results are to be viewed as suggestive only - larger-scale investigations, outside the scope of the work discussed here, are needed to verify fully our hypothesis. Some more structured experiments are discussed in chapter 6.

This partially addresses the problem of compensating for the crucial equivalent effect in translations obtained using a single small-scale dictionary without increasing the swamping effect. In our *Combined* translations, there was an increased swamping effect, but were we to repeat the less ambiguous equivalents only, we would still observe an improvement in retrieval performance. This means that combining a number of similar dictionaries can improve retrieval, but only if the ambiguity of repeated equivalents is taken into consideration.

We shall now examine some aspects a completely different dictionary characteristic that can affect retrieval performance, namely that of dictionary coverage rate prior to coverage compensation.

5.4 Coverage Rate and Retrieval Performance

Here, we take a closer look at our *coverage compensation* procedure. (Coverage compensation, described in chapter 3, involves adding entries from a larger scale dictionary, in our case, *LargeNoRep*, to a smaller-scale dictionary for terms which do not have an entry in the smaller-scale dictionary).

We ask, what level of coverage does a small-scale dictionary have to provide *prior* to the application of coverage compensation, in order for the retrieval performance of query translations obtained from it *after* coverage compensation using *LargeNoRep* to be significantly higher than that recorded for translations obtained using *LargeNoRep* alone?

To answer this question, we created a series of *reduced coverage* versions of *SGemNoRep*, where we reduced its coverage by 10% each time, and examined the results. Coverage was reduced by removing the least ambiguous terms first. This was to mirror the effect of using smaller and smaller scale dictionaries - the smaller the scale, the less likely a dictionary was to contain unusual and therefore less common terms. For example, one is more likely to find an entry for *world* is more likely than one for *deforestation* in a small-scale dictionary.

We found that reducing the coverage rate had little effect on performance until coverage was reduced

to levels below 40%. Beyond 40%, a drop in performance was observed each time, until we reached 10%, where there was no difference in performance between the translations obtained using a coverage-compensated *SGemNoRep* and the corresponding *LargeNoRep* translations.

Knowing the minimum coverage rate needed for a new small-scale dictionary to improve the retrieval performance of query translations on its addition to a CLIR system is useful if one is planning on inserting one into an existing CLIR system which already contains a larger-scale dictionary. Alternatively, if there is a choice of small-scale dictionaries, this can help determine which would be the best candidate for insertion into the system. In addition, when porting an existing system to a new subject domain, it allows us to ascertain how effective any existing small-scale system dictionaries would be when added to the ported system, based on their coverage of the new domain.

We concluded that a coverage rate of only 20% was needed for a small-scale dictionary to improve the performance of associated query translation when it was introduced into a CLIR system which already employed a larger-scale dictionary, providing that the terms covered by the small-scale dictionary were the more ambiguous terms in our test query set. This is because when a larger-scale dictionary is used to translate highly ambiguous terms, a great many equivalents, many of which have nothing to do with the meaning of the original query, are added to the query translation. Cutting down on the number of translations provided for the most ambiguous terms is more important than cutting the number of translations for terms with 2 possible translations from 2 equivalents to 1.

Since the use of a small-scale dictionary to look up equivalents for a given term instead of *LargeNoRep* effectively constitutes a type of equivalent selection, our results also imply that an effective equivalent selection technique need only concentrate on the 40% most ambiguous terms in our query set and not bother with the others. How this would generalise to any arbitrary query set is yet to be determined. The following subsections describe in more detail the steps followed the experiments described above.

5.4.1 Creating Reduced Coverage Versions of *SGemNoRep*

We created a series of *reduced coverage* versions of *SGemNoRep* as follows. The coverage-compensated *SGemNoRep* employed in our experiments so far has a coverage rate of 100% of the terms in our test query set. We called this dictionary *SGemNoRep100*. Prior to the application of coverage compensation, our original *SGemNoRep* as derived from the Collins Gem Pocket Dictionary had a coverage rate of 85%. This is *SGemNoRep85*. Then, we created a list of all of the terms for which *SGemNoRep85* contained an entry and sorted this list in ascending order of ambiguity. We used this list to create further new dictionaries with lower levels of coverage each time. *SGemNoRep80* was obtained from *SGemNoRep85* by taking the top 5% terms (calculated as 5% of 385, the total number of terms in our test query set) from the list described above and *deleting* their entries from *SGemNoRep85*. *SGemNoRep70* was obtained from *SGemNoRep80* by deleting a further 10% of terms (remember, 10% of 385) using the next 10% of terms in the list, and so on, down to *SGemNoRep10*. Coverage compensation using *LargeNoRep* was then applied to each of these new dictionaries and a set of query translations obtained for each new coverage-compensated dictionary.

We posed the following hypothesis:

5.4.2 Hypothesis 5C: The More Ambiguous the Term, the More it Benefits Retrieval Performance to Reduce the Number of Equivalents Provided for it by Employing a Small-Scale Dictionary to Translate it

For this hypothesis to be verified, we would need to observe a drop in R-Prec performance as the coverage rate of *SGemNoRep* is reduced, with each successive drop being greater than the one before, until performance drops to the level of the *LargeNoRep* translations. This latter point is where we deem the coverage rate of the smaller-scale dictionary to be too low to have any beneficial effect on retrieval performance. The results of Add-All-Equivalents translation of our test query set using each of our new coverage-compensated dictionaries are displayed in Figure 5.11, showing the R-Prec values and the comparative drop in performance between one set of translations and the set of translations immediately above it. (AvP values were also available, but are not quoted here for reasons of space). The results of significance tests based on these runs are shown in Figure 5.12.

Run	Unweighted	Drop	Weighted	Drop
SGemNoRep100	10	-	22	-
SGemNoRep85	9	1	21	1
SGemNoRep80	10	-1	22	0
SGemNoRep70	10	0	22	0
SGemNoRep60	10	0	22	0
SGemNoRep50	11	-1	22	0
SGemNoRep40	9	2	21	1
SGemNoRep30	7	2	19	2
SGemNoRep20	6	1	17	2
SGemNoRep10	7	-1	16	1
LargeNoRep	6	1	15	1

Figure 5.11: Progressively Reducing the Coverage Rate of *SGemNoRep*

Runs Compared	UnW RP	UnW AvP	W RP	W AvP
100% v. 60%	0.049	0.048	0.453	0.002
100% v. 50%	0.072	0.023	0.306	0.009
100% v. 40%	0.018	0.001	0.039	0.001
40% v. 30%	0.001	0.007	0.004	0.001
30% v. 20%	0.199	0.004	0.0	0.001

Figure 5.12: Significance Tests - Probability of the Null Hypothesis

5.4.3 Significance of Hypothesis 5C

Verification of this hypothesis would show that most energy at the equivalent selection and S-weighting stage should be expended on devising selection techniques for the most ambiguous terms. This would mean that the strategies applied to a given term or its equivalents could be targeted according to the term's degree of ambiguity.

5.4.4 Results and Partial Verification

The results are not as clear as one would have hoped, but nevertheless, a pattern emerges. The removal of small-scale dictionary entries for the 60% least ambiguous terms in our test query set did not affect retrieval performance significantly. This indicates that substituting entries from *LargeNoRep* for the *SGemNoRep* entries for these terms did not affect retrieval. Since the 60% least ambiguous terms tended to have few equivalents, ranging between 1 and 3, we concluded that most effort should be concentrated on reducing the number of equivalents provided for the most ambiguous terms.

As expected, beyond reduction to 40%, things change. Retrieval performance is significantly lower (3 tests out of 4 show significance at 95%) than for the full 100% run. In addition, for each successive drop in coverage rate, for example, from 40% to 30%, performance drops significantly. This suggests that reduction of the number of equivalents added to the query translation for these 40% most ambiguous source-language query terms is important for retrieval performance.

5.4.5 Concluding Remarks on Hypothesis 5C

Thus hypothesis 5C has only been verified partially, but its investigation has yielded valuable insights. Firstly, equivalent selection algorithms need to concentrate on the 40% most ambiguous terms in our test query set only. Secondly, performing good equivalent selection is equally important for the terms in the 40th to 30th percentile as regards ambiguity as for the 10% most ambiguous terms. Finally, a small-scale dictionary needs to provide a coverage rate of 20% or more to give rise to an improvement in query translation performance when combined with *LargeNoRep* using our coverage compensation procedure. This allows us to assess the benefit to retrieval performance of the insertion of a given small-scale dictionary into an existing CLIR system without having to perform the insertion and integration of the dictionary first - useful when deciding which dictionaries to add to an existing system to improve

its performance, or when porting a system to a new subject domain to ascertain which of the existing system dictionaries would still be an effective part of the system despite the change in domain.

5.5 Conclusions

In this chapter, we examined the effect on retrieval performance of associated query translations of a number of dictionary characteristics other than scale. In particular, we looked at the effect of small variations in query translation content between a number of dictionaries of similar scale derived from similar printed bilingual dictionaries aimed at the same market, and how these variations affected performance. This led us to revisit the issues of dictionary combination in query translation and of equivalent repetition, as well as the link between equivalent repetition and equivalent ambiguity and retrieval performance. We also discussed the effect on performance of the coverage rate of small scale dictionaries combined with larger-scale ones as described in our coverage compensation procedure.

We reached the following conclusions:

- Retrieval performance was similar for three different dictionaries of similar scale and content.
- No dictionary, no matter how small-scale, is immune to the crucial equivalent effect - minor variations in between two translations of a given query can result in large differences in retrieval performance. It is not possible to tell a priori which query translations will be affected by the crucial equivalent effect.
- The reduction of the crucial equivalent effect brought about by combining all three dictionaries during translation was not sufficient to counter the resulting increased swamping effect.
- If one retains the equivalent repetition that results from dictionary combination during translation, performance can be improved using our retrieval engine. However, only some queries benefit from this improvement.
- Only repetition of the less ambiguous equivalents results in improved performance for query translations obtained using a combination of dictionaries (assuming a similar query processing strategy to that implemented by our retrieval engine is employed).
- Less ambiguous equivalents do not appear to be less frequent in the document collection. This flies in the face of current monolingual IR knowledge and warrants further investigation.
- Our retrieval engine R-weighting strategy can benefit from the implicit additional S-weighting introduced by repetition of the less ambiguous equivalents.
- Dictionary combination is therefore an effective way of handling the crucial equivalent effect if the less ambiguous equivalents only are repeated in the query translation.
- A small-scale dictionary must provide a rate of coverage prior to coverage compensation being applied of at least 20% for an improvement in retrieval performance using a larger-scale dictionary alone is recorded when the small-scale dictionary is combined with a larger-scale dictionary using our coverage compensation procedure.
- There is no need to apply any equivalent selection strategies to the 60% least ambiguous terms in our test query set.
- Equivalent selection strategies should concentrate on reducing the number of equivalents provided for the 40% most ambiguous terms in our test query set.

Some questions remain to be answered. Our results have shown that we do not need to worry about applying equivalent selection techniques to the 60% least ambiguous terms in our test query set. But how does this translate to the general case? Do we calculate the degree of ambiguity of the most ambiguous term in this 60%, and use this as an arbitrary threshold for all other possible queries? Do we keep a database of past queries and calculate a similar threshold based on that data? Or do we simply ignore the 60% most ambiguous terms in each query, irrespective of content? And should we delete some highly ambiguous terms or equivalents altogether?

The next chapter investigates some more structured experiments regarding the S-weighting of more and less ambiguous terms and equivalents in queries and query translations.

Chapter 6

Equivalent T-Weighting and Term Q-Weighting

In chapters 4 and 5, we looked at how different characteristics of our CLIR dictionaries affected the retrieval performance of associated query translations. We saw how a good knowledge of dictionary characteristics could be used to improve query translation retrieval performance. One of the characteristics examined was the phenomenon of equivalent repetition within dictionary entries and how the resulting repetition of equivalents and inherent implicit application of S-weights in the query translation could both benefit and harm retrieval performance. In particular, we saw that the repetition of, and therefore increasing the S-weight applied to, less ambiguous equivalents tended to result in improved retrieval performance, whereas repeating more ambiguous equivalents tended to have an opposite effect.

In this chapter, we exploit this result to apply explicit S-weights to query terms before translation, which we call Q-weighting (see chapter 3), and to equivalents after translation, which we term T-weighting, based on their *degree of ambiguity*. As in previous chapters, the degree of ambiguity of a query term is defined as the number of distinct equivalents listed for it in *LargeNoRep*. The degree of ambiguity of an equivalent is the number of distinct possible translations provided for it by the French-English portion of the Collins-Robert Unabridged Dictionary (*LargeNoRep* was derived from the English-French portion of this dictionary). This last measure was calculated by hand as an electronic version of this portion of the dictionary was not available to the experimenter. We may classify the application of T-weights to query terms as a form of pre-translation query modification (stage 1 of dictionary-based CLIR - see chapter 2), and the application of Q-weights to equivalents after translation as belonging to the equivalent selection and S-weighting stage (stage 3 of the process).

Finally, we took the best performing method resulting from all of the investigations reported in this thesis and combined it with the best performing term and equivalent Q- and T-weighting strategies tested here. **Performance for this set of "best" query translations was only slightly less than the best results recorded in the literature for dictionary-based query translation using complex processing strategies. This demonstrated that it is possible to obtain a very good level of retrieval performance without using large-scale linguistic resources and without performing time-consuming and expensive processing of the retrieval collection**.

A number of different tests were used for calculating significance in this chapter, as the paired T-test methodology employed in previous chapters was felt to be too conservative. In addition to performing paired T-tests on data derived from the "raw" AvP and R-Prec evaluation data as described in chapter 3, we also carried out Sign Tests and in some cases Wilcoxon Signed Rank Tests, directly on the AvP and R-Prec results for each query.

We now present the different Q- and T-weighting strategies investigated.

6.1 Types of Q- and T-weighting Investigated

We have grouped our approaches into the following categories:

- Assigning a T-weight of 0.0 to equivalents whose *degree of ambiguity* exceeded a certain threshold, and of 1.0 to all others. (This amounted to deleting more ambiguous equivalents from the query

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
DeleteAbove1	10	19	8	14	10	19
DeleteAbove9	8	20	10	17	10	22
DeleteAbove11	8	21	10	19	10	22
DeleteAbove13	9	21	11	19	11	24
DeleteAbove15	9	21	11	19	11	24

Figure 6.1: Deletion of Equivalents of Degree of Ambiguity Greater than Threshold

translation).

- Applying a T-weight greater than 1.0 to equivalents whose degree of ambiguity was less than a certain threshold, and of 1.0 to all others. This boosted the retrieval-time weight applied to these less ambiguous equivalents without eliminating the more ambiguous equivalents completely.
- Assigning a Q-weight of 0.0 to *query terms* whose degree of ambiguity exceeded a certain threshold, and of 1.0 to all others (i.e. deleting more ambiguous terms prior to translation).
- Applying a Q-weight greater than 1.0 to terms whose degree of ambiguity was less than a certain threshold, and of 1.0 to all others.

The *CombinedNoRep* translations discussed in chapter 5 were used as a base for our experiments concerning equivalent Q- and T-weighting.

In the case of term Q-weighting, we applied the same translation algorithm as for the *CombinedNoRep* translations after the Q-weights had been applied. Any Q-weight applied to a given term was applied to all of its equivalents after translation. We remind the reader that as before, all experiments were run twice, once with R-weighting enabled, and once with it disabled. This was to observe the effect of the Q-weighting strategies on retrieval performance both in conjunction with R-weighting and in isolation.

6.2 Applying a T-Weight of 0.0 to More Ambiguous Equivalents

A T-weight serves as a multiplier of the R-weight calculated by the retrieval engine. Assigning a T-weight of zero effectively results in a total weight of zero being assigned to that equivalent at retrieval time, irrespective of the R-weight calculated by the retrieval engine, as a result of the manner in which our retrieval engine processes queries (see chapter 3). Here, we look at reducing the total number of equivalents in the *CombinedNoRep* translations by assigning a T-weight of 0.0 to those equivalents in this set of query translations whose degree of ambiguity *exceeds* a given threshold N, for a number of values of N.

6.2.1 New Sets of Query Translations

We obtained a new set of T-weighted query translations from the *CombinedNoRep* translations for each value of $N = 1, 2, 3, 4, 5, 7, 9, 11, 13$ and 15 by deleting (assigning a T-weight of 0.0) from the *CombinedNoRep* translations any equivalent whose degree of ambiguity exceeded N. The results of running these new sets of query translations on the retrieval collection are displayed in full in Appendix 6.(i), with an abbreviated set of results in Figure 6.1. The probability of the null hypothesis for associated significance tests (Paired T-test and Sign Test) is also displayed in Appendix 6.(i).

6.2.2 Results

The results show that removing equivalents did not significantly improve performance over the *CombinedNoRep* translations for values of N less than 11. However, using the Paired T-Test, retrieval performance was significantly better for the sets of query translations obtained for $N = 11, 13$ and 15 than for the

Degree of Ambiguity	Num Equivalents
1	291
2-3	207
4-5	149
6-7	103
8-9	65
10-11	50
12-13	45
14-15	21

Figure 6.2: Equivalents in CombThreeNoRep query translations in each ambiguity range

CombinedNoRep translations, and the same was true for $N = 13$ and $N = 15$ when we looked at significance using the Sign Test. This indicated that was it a good idea to assign a T-weight of 0.0 to highly ambiguous equivalents, i.e. those whose degree of ambiguity exceeded 11. The negative effect on performance of deleting important equivalents outweighs any benefit due to a reduced swamping effect for values of N less than 11 - to such an extent where $N = 1$ that performance is in fact significantly worse using the Paired T-Test than for the *CombinedNoRep* translations. Therefore, we should limit our activities as regards equivalent deletion (applying a T-weight of 0.0) to those equivalents of degree of ambiguity greater than 11.

In addition, we can see in Figure 6.2 that the number of equivalents whose removal benefited retrieval performance was actually quite small - around 10% or less of all unique equivalents in the *CombThreeNoRep* query translations. This demonstrates that a small number of highly ambiguous equivalents can have enough of a swamping effect to cause significant problems for retrieval. Therefore, work aimed at reducing the swamping effect after query translation would do well to concentrate on the more ambiguous translation equivalents. We can draw a parallel between this conclusion and that discussed in chapter 5, where we found that concentrating attention on the 40% most ambiguous it terms in our test query set yielded the best results.

Since the absolute value calculated for the degree of ambiguity of any equivalent is dependent on the dictionary used to calculate it, further experimentation is needed using a variety of different dictionaries to calculate this measure before this result can be applied to the general case. This was outside the scope of this thesis.

6.3 Applying Additional T-Weights to Less Ambiguous Equivalents

Above, we sought to reduce the swamping effect associated with more ambiguous equivalents by applying a T-weight of 0.0 to those held to be highly ambiguous, and assigned the default T-weight of 1.0 to all others. Here, we look at the other end of the scale, where we apply T-weights greater than 1.0 to *less* ambiguous equivalents - those whose degree of ambiguity is equal to or less than a given threshold N. We saw in chapter 4 and 5 that repeating less ambiguous equivalents in the query translation could lead to improved retrieval performance. Assigning a T-weight greater than 1.0 to an equivalent has the same effect as repeating it due to the manner in which our system processes queries. For example, assigning a T-weight of 2.0 to an equivalent is the same as repeating the equivalent once, whereas a T-weight of 4.0 is similar to adding three extra occurrences of the given equivalent to the query translation.

We investigated assigning a T-weight greater than 1.0 to less ambiguous equivalents, using an arbitrary threshold as for the equivalent deletion experiments above, and a fixed T-weight. We performed experiments for several different threshold values (N) and three different higher T-weights.

6.3.1 New Sets of Query Translations

We obtained three new sets of query translations from the *CombinedNoRep* translations for each threshold value N where $N = 1, 2, 3, 4, 5, 7, 9, 11, 13$ and 15. In the first set of translations for each value of N, we assigned a T-weight of 2.0 to those equivalents whose degree of ambiguity was less than or equal to N. In the second and third set of query translations for each value of N, T-weights of 3.0 and 4.0 were

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
T-weight = 2.0						
CombinedNoRep	8	20	10	18	10	21
Threshold1	13	25	13	20	14	25
Threshold2	12	23	13	21	12	24
Threshold3	12	22	12	20	13	24
Threshold4	11	21	11	18	12	23
Threshold5	10	21	11	18	11	22
Threshold7	9	20	11	18	11	22
Threshold9	9	20	11	19	11	22
Threshold11	9	21	10	19	11	22
Threshold13	9	21	11	19	11	23
Threshold15	9	21	10	19	11	23
T-weight = 3.0						
Threshold9	9	21	11	19	11	22
Threshold11	10	21	11	19	12	23
Threshold13	9	21	11	19	11	24
Threshold15	9	21	10	19	11	23

Figure 6.3: Applying a T-weight Greater than 1.0 to Equivalents of Degree of Ambiguity Below or Equal to Threshold

applied to these equivalents. All other equivalents were assigned the default T-weight of 1.0. No attempt was made at this stage to combine this strategy with the equivalent deletion techniques described above. This gave us 30 new sets of query translations. Results for running these translations on the retrieval collection are given in full in appendix 6.(i), in tables 4, 7 and 10, and associated significance test values (paired T-Tests and Sign Tests) in tables 5, 8, 6, 9, 11 and 12. Abbreviated results are available in Figure 6.3.

6.3.2 Results

Assigning a T-weight of 2.0 to less ambiguous equivalents did not result in any significant improvement in performance over the *CombinedNoRep* translations for either significance test employed. (Although results for AvP for values of $N = 9, 11, 13$ and 15 were significantly better for the Paired T-Test, the corresponding R-Prec results were not significantly better, therefore, we cannot conclude significance from these results, and the Sign test did not show significance anywhere). Therefore, either assigning a T-weight to less ambiguous equivalents is not an effective strategy, or the T-weight assigned, 2.0, was not high enough.

On applying a T-weight of 3.0 to equivalents whose degree of ambiguity was less than N, the only values of N for which the thresholded query translations obtained significantly better performance scores using both the Paired T-tests and the Sign Test than the *CombinedNoRep* translations were 11, 13 and 15. Results for T-weights of 4.0 were significantly better for $N = 13, 15$ only, for both tests.

In addition, for a given value of N, no one T-weighting strategy's retrieval performance was significantly superior to another's for the same value of N (using the Paired T-Test), provided both were significantly better than the *CombinedNoRep* translations. Furthermore, none of these three T-weighting strategies resulted in retrieval performance scores that were significantly different from those observed in the previous section where equivalents of a degree of ambiguity greater than N were assigned a T-Weight of N, again, using the Paired T-Test.

Therefore, using a threshold value of 11, we can either assign a T-weight of 0.0 to equivalents whose degree of ambiguity exceeds the threshold, or we assign a T-weight of 3.0 or greater to equivalents of a degree of ambiguity less than or equal to the threshold. Since assigning a T-weight of 0.0 is the simpler strategy to implement, we retained this as the equivalent T-weighting strategy of choice. Once more, the absolute value of the threshold for the general case needs to be determined by further experiments, which were outside the scope of this thesis.

Now we look at a way modifying the query prior to translation, by applying Q-weights to *query terms*.

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
DeleteAbove1	13	23	9	16	13	23
DeleteAbove2	15	27	11	20	14	27
DeleteAbove3	16	27	13	20	16	27
DeleteAbove4	16	26	13	20	16	27
DeleteAbove5	14	24	12	19	14	26
DeleteAbove7	14	25	12	20	15	27
DeleteAbove9	12	25	12	21	13	27
DeleteAbove11	11	25	12	21	13	26
DeleteAbove13	11	24	11	20	12	25
DeleteAbove15	9	21	11	19	11	22

Figure 6.4: Deletion of Terms of Degree of Ambiguity Greater than Threshold

6.4 Applying a Q-Weight of 0.0 to Source-Language Query Terms

Query term Q-weighting is where we assign a S-weight to some of the source-language query *terms* before any translation is carried out. After translation, any Q-weight assigned to a given source-language query term is assigned to every equivalent of that term obtained during translation. The aim is to preempt the damaging effect of term ambiguity by reducing the influence of equivalents which were obtained for more ambiguous terms. Term Q-weighting can be implemented along with any equivalent T-weighting method, where the term and equivalent weights are multiplied to obtain each equivalent's final S-weight.

This section concerns the outright deletion terms from the source-language query prior to translation. This amounts to assigning a Q-weight of 0.0 to some terms and of 1.0 to others, due to the manner in which our system processes queries. The idea here is that most of the extra, unwanted equivalents in a typical query translation result from the profusion of equivalents supplied by the dictionary for a small number of highly ambiguous terms in the original query. These many equivalents may be eliminated by deleting the terms responsible for their presence from the query before any translation is performed. In addition, the assumption is being made that highly ambiguous terms will not contribute a great deal to the statement of the user information need, a fact which is reflected in the low term weights calculated by mainstream retrieval engines for highly frequent terms in a monolingual setting. This should allow us to eliminate such terms without unduly harming the retrieval performance of the resulting query translations.

6.4.1 New Sets of Query Translations

We constructed several new versions of our 80-query English-language test query set. For each value of N in $N = 1, 2, 3, 4, 5, 7, 9, 11, 13, 15$, we obtained a new set of source-language queries where any term of degree of ambiguity greater than N was removed. The degree of ambiguity of a given query term was defined as the number of distinct equivalents provided for it by *LargeNoRep* (see above). This measure, unlike the degree of ambiguity of an equivalent, was calculated automatically by looking up *LargeNoRep*. *LargeNoRep* was always employed irrespective of the dictionaries used in the rest of any experiment as we want to ensure consistency of definition across experiments. These new sets of queries were translated using the same algorithm used to obtain our *CombinedNoRep* translations (see chapter 5). The results of running these new sets of query translations on the retrieval collection are displayed in Figure 6.4 and associated significance values for the Paired T-Test and the Sign Test are shown in Appendix 6.(ii) in Figures 13 and 14.

6.4.2 Results

We see that, contrary to what one might expect from examining the retrieval performance scores displayed in Figure 6.4, there was no significant difference in underlying performance between any of the thresholded runs above and the *CombinedNoRep* translations, for either test employed. There are two possible reasons

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
Q-weight = 2.0						
Threshold1	12	24	13	21	15	26
Threshold2	14	27	14	22	16	29
Threshold3	15	27	14	21	17	28
Threshold4	15	26	14	21	17	28
Threshold5	14	25	12	21	15	27
Threshold7	13	25	13	21	15	27
Threshold9	11	25	13	21	12	27
Threshold11	10	25	12	21	12	26
Threshold13	10	24	12	20	11	25
Threshold15	9	22	11	19	10	22
Q-weight = 3.0						
Threshold1	14	24	14	19	17	25
Threshold2	17	28	16	21	19	29
Threshold3	17	28	15	21	19	29
Threshold4	17	27	15	21	18	28
Threshold5	15	26	13	21	16	27
Threshold7	14	26	14	21	15	28
Q-weight = 4.0						
Threshold3	18	27	16	21	20	28

Figure 6.5: Applying a Q-weight Greater than 1.0 to Terms of Degree of Ambiguity Below or Equal to Threshold

for this. Firstly, the differential in Q-weight between less and more ambiguous terms may not have been sufficiently large. We investigate this possibility in the next section. Secondly, applying different Q-weights to terms prior to translation may not be a useful strategy irrespective of the Q-weights assigned. Should we not find any significant improvements in the next section, where we assign Q-weights greater than 1.0 to less ambiguous terms, we can conclude the latter.

6.5 Higher Q-Weighting of Less Ambiguous Terms

As in our experiments concerning equivalent T-weighting above, we investigated applying Q-weights greater than 1.0 to source-language query terms whose degree of ambiguity was less than or equal to a given threshold N.

6.5.1 New Sets of Query Translations

We obtained three new sets of source-language queries for each value of N, $N = 1, 2, 3, 4, 5, 7, 9, 11, 13$ and 15 - one set, where a Q-weight of 2.0 was applied to each term of degree of ambiguity greater than or equal to N, one where the Q-weight applied was 3.0, and one with a Q-weight of 4.0 being applied. This gave us 30 new source-language query sets in all. These were then all translated using the same algorithm employed to obtain the *CombinedNoRep* translations and run on the retrieval collection. Q-weights were conserved during translation as described in the previous section. Full results for these runs are displayed in Appendix 6.(ii) in Figures 15, 18 and 21, and associated significance values in Figures 16,17, 19, 20, 22 and 23. Abbreviated results are displayed in Figure 6.5 here.

6.5.2 Results - Assigning a Q-weight of 2.0

Where we assigned a Q-weight of 2.0 to terms whose degree of ambiguity was less than or equal to N and translated the resulting query set, retrieval performance was significantly better than for the *CombinedNoRep* translations for all values of N using the Paired T-Test and for all values of N bar $N = 13, 15$ using the Sign Test. This contrasted with the results reported in the last section (assigning

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
Stepped2	14	27	14	23	16	28
Stepped3	16	28	15	23	18	30
Stepped4	16	29	16	23	17	31
Stepped5	15	29	15	23	16	31
Stepped7	14	27	15	22	15	30
Stepped9	13	27	14	22	15	29
Stepped11	13	26	14	21	14	28
Stepped13	12	25	14	21	14	27
Stepped15	12	25	14	21	13	26

Figure 6.6: Applying Q-Weights According to Step Function

a Q-weight of 0.0 to ambiguous terms) which were not significantly better than the *CombinedNoRep* translations' retrieval performance scores. The influence of equivalents associated with more ambiguous terms may have been reduced without eliminating these equivalents altogether, thus resulting in better performance where one of these ambiguous terms' equivalents turned out to be crucial for retrieval performance for some of the query translations. Furthermore, retrieval performance where the Q-weight was 2.0 and $N = 1, 3$ or 7 was significantly better than for other values of N.

6.5.3 Results - Assigning a Q-weight of 3.0 or of 4.0

For values of N below 13, applying a Q-weight of 3.0 to terms whose degree of ambiguity was less than or equal to N resulted in retrieval performance which was significantly better than that of the *CombinedNoRep* translations for both the Paired T-Test and the Sign Test. Similar results were observed where a Q-weight of 4.0 was applied in identical circumstances for values of N less than 13. Remember that where Q-weights of 2.0 were applied, significantly better performance was recorded for *all* values of N tested. None of the Q-weighted translations which were significantly better than the corresponding *CombinedNoRep* translation, whether the Q-weight was 2.0, 3.0 or 4.0 was significantly better than any other such translation.

These results indicate that the magnitude of the differential between the higher and lower Q-weights assigned (2.0 v. 1.0. 3.0 v. 1.0 etc) was not as important as ensuring that a differential was present. Therefore, we chose assigning a Q-weight of 2.0 to all terms of degree of ambiguity less than or equal to 11 as our term Q-weighting method of choice.

6.5.4 Stepped Q-Weighting of Terms

Following on the results of our paired T-tests, we obtained some new sets of source-language queries which applied a "stepped" Q-weight application function. We assigned a weight to each term based on the difference between N and the degree of ambiguity:

$$(S - weight) = N - DegAmb(term_i) + 1$$

if the degree of ambiguity of the $term_i$ was less than or equal than N, and 1.0 was assigned otherwise. A new set of queries was obtained for each value of N, $n = 2, 3, 4, 5, 7, 9, 11, 13$ and 15. This is an ad hoc formula, and is not based on any theory, and so it should be viewed as a preliminary investigation of graduated Q-weighting methods only.

The resulting sets of queries were then translated using the same algorithm employed to obtain the *CombinedNoRep* translations. The results of running these query translations on the retrieval collection tests are displayed in Figure 6.6 and associated significance tests in Appendix 6.(iii) in Figure 24. As there was no motivation in our Sign Test results for a stepped approach to query Q-Weighting, we did not carry out Sign tests here.

All of our Stepped runs performed significantly better than the *CombinedNoRep* translations using the Paired T-Test. In addition, our new sets of query translations for values of N from 2 to 5 performed significantly better than those obtained for other values of N. However, none of these runs performed significantly better than assigning a uniform Q-weight of 2.0 to terms of degree of ambiguity less than

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
FinalCombined	15	27	15	18	16	28
DeleteAbove9	12	25	12	21	13	27
Thresh3W2.0	14	27	14	22	16	29
Stepped4	16	29	16	23	17	31
CombinedNoRep	8	10	10	20	18	21
Combined	10	22	15	27	12	24
AutoVerySmall	12	25	11	21	13	26
Teensy	11	24	11	19	12	25
LargeNoRep	5	13	6	12	5	15
Perfect Dictionary	16	31	14	25	15	31
French Human	18	34	18	28	17	33

Figure 6.7: Comparative Results for Final S-weighting Combination

Paired T-Test	UnW AvP	W AvP	UnW RP	W RP
FinalCombined v. CombinedNoRep	0.0	0.0	0.46	0.57
FinalCombined v. AutoVerySmall	0.07	0.70	0.08	0.33
FinalCombined v. Teensy	0.0	0.01	0.0	0.0
Sign Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
FinalCombined v. CombinedNoRep	0.0	0.0	0.0	0.0
FinalCombined v. AutoVerySmall	0.11	0.0	0.0	0.0
FinalCombined v. Teensy	0.05	0.01	0.22	0.03
Wilcoxon Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
FinalCombined v. CombinedNoRep	0.0	0.0	0.0	0.0
FinalCombined v. AutoVerySmall	0.0	0.0	0.06	0.01
FinalCombined v. Teensy	0.27	0.02	0.01	0.04

Figure 6.8: Final S-weighting Combination - Significance Tests

or equal to 3 (*Thresh3W2.0*), again using the Paired T-test. Since assigning the uniform Q-weight is the simpler method, we have retained it as our term Q-weighting technique of choice.

In the next section, we combined all of the insights obtained in previous chapters with the results quoted above.

6.6 Final Combinations and Comparisons

Here we combined the best performing methods from all of the experiments described in chapters 4 to 6 of this thesis. We attempted to obtain "best" absolute and relative performance values for our accumulation of simple methods. We wanted to see if our accumulation of simple methods based on dictionary information only could rival the more complex strategies discussed in the literature for dictionary-based CLIR.

A new set of query translations, *FinalCombined*, was obtained by combining the deletion of equivalents of degree of ambiguity equal to or greater than 10 with the application of a uniform S-weight of 2.0 to terms of degree of ambiguity less than or equal to 3. Retrieval performance results for this new set of query translations, along with some past results, are displayed in Figure 6.7, and associated significance tests in Figure 6.8.

The Sign Test demonstrated significance (values of 0.0) for all four measures between the *Final Combined* and the *CombinedNoRep* translations, and between the *Final Combined* and *AutoVerySmall* translations, as did the Wilcoxon Signed Rank test (as these results are important, we carried out several different tests). The *Teensy* translations were almost significantly different, but not quite, for both the Sign and Wilcoxon tests, and were considered significantly different by the Paired T-Test.

This indicates that our *FinalCombined* translation method can be viewed as slightly better than either the translations obtained using the *AutoVerySmall* or the *Teensy* dictionary, the best results we obtained in previous chapters. Therefore we can consider it the method of choice for query translation for CLIR

using our system.

Further experiments are necessary to investigate whether the above results hold for additional collections, query sets and language pairs. These were outside the scope of this project.

6.7 Final Outcome of Project

We stated in chapter 4 that no translation method of ours could expect to outperform the Perfect Dictionary translations. Our *FinalCombined* translations above obtained, with retrieval engine
R-weighting enabled, 79% and 91% of the monolingual upper bound (French Human translations) AvP and R-P performance respectively, and 87% and 97% of the Perfect Dictionary AvP and R-P performance. This compares well with the best in the field - Ballesteros and Croft reported results of 85% of the monolingual upper bound, for example [10] (although the usual caveats must be borne in mind: different IR systems were used, and the document collections were slightly different - see chapter 2). Thus, we have demonstrated that applying a combination of simple methods by exploiting dictionary characteristics can result in effective cross-language information retrieval without resorting to involved processing of the retrieval collection and without recourse to an outside resource such as an MT engine or a parallel corpus.

6.8 Conclusions

This chapter concentrated on equivalent and term S-weighting methods that built on the insights gained in our investigations of the effect of equivalent repetition on query translation retrieval performance to assign S-weights based on ambiguity information obtained from the dictionary only. We also put the final combination of our efforts to the test, to see how it measured up against the results quoted in the literature for more involved equivalent selection and S-weighting methods, such as those described by Ballesteros and Croft [10].

We made the following discoveries:

- Deleting (assigning a Q-weight of 0.0 to) equivalents of a degree of ambiguity greater than or equal to 11 from the query translation led to significantly better retrieval performance.
- Assigning higher Q-weights to less ambiguous equivalents while leaving more ambiguous equivalents with the default Q-weight of 1.0 was not advantageous to the retrieval performance of associated query translations, contrary to expectations.
- Deleting source-language query *terms* from the query prior to translating it using the algorithm employed to obtain the *CombinedNoRep* translations was not particularly effective.
- Assigning uniform higher T-weights to less ambiguous terms prior to translation while assigning the default T-weight of 1.0 to all other terms gave rise to significant improvements in retrieval performance for threshold values under $N = 11$.
- Applying a stepped Q-weighting function to terms was not significantly better than assigning a uniform weight of 2.0 to terms of degree of ambiguity less than or equal to the threshold.
- The final combination of all the insights obtained during this project obtained retrieval performance scores with retrieval engine R-weighting enabled of 79% and 91% of the monolingual upper bound (French Human translations) AvP and R-P performance scores respectively, and 87% and 97% of the Perfect Dictionary AvP and R-P performance scores. This was significantly better than our results obtained using dictionaries of scale close to 1.0 and compares well with the best in the field.
- **Therefore, applying a combination of simple methods by exploiting dictionary characteristics is an effective and successful strategy for cross-language information retrieval**.

The next chapter summarises all the work presented in this and previous chapters, describing the insights into the CLIR process obtained during the course of this project and explains why these insights are significant for CLIR.

Chapter 7

Conclusions and Future Work

The provision of more and more information and text in an electronic format worldwide underpins a growing demand for effective, robust CLIR. However, as we saw in chapter 2, substantial bottlenecks remain in the form of the resources exploited by existing approaches to CLIR which mean that implementing CLIR is still a relatively expensive and time-consuming business. For example, although machine translation (MT) of the user request without further processing is the most effective CLIR method reported in the literature, it relies on the availability of a commercial MT engine for the relevant language pair. Corpus-based bag-of-words query term mapping systems also depend on a hand-crafted resource, an aligned bilingual parallel corpus. Finally, the best performing dictionary-based strategies for CLIR reported in the literature rely on heavy-duty processing of the retrieval collection to extract information such as co-occurrence frequencies to select the "correct" translation equivalent for each term in the bag of words query derived from the user request.

What then, can be done when a suitable MT engine or parallel corpus is not available, or is too expensive, and intensive processing of the collection is not practical (for example, when the collection is updated daily or hourly and is very large)? For CLIR technology to become widespread, it must be possible to implement an effective CLIR strategy in ALL situations, at minimal cost.

Our work looked at CLIR using information from the system dictionary only. We focused on examining how different characteristics of the system dictionary affected retrieval performance in a series of carefully controlled experiments investigating the effects on performance of certain characteristics in isolation. We found that by exploiting the insights gained during this examination of dictionary characteristics, a level of retrieval performance comparable to the best reported in the literature could be achieved.

Converting an electronic dictionary to a format which can be employed directly by a CLIR system is non-trivial but takes much less time and effort than developing an MT engine or aligning a bilingual corpus. The absence of intensive processing of the retrieval collection from our approach means that considerably less computing power is needed. Therefore, this work can help make CLIR more accessible to new language pairs by making the development of a working system faster, easier and cheaper.

7.1 Presentation of Findings

The following sections the findings of this research, grouped according to dictionary characteristic investigated:

- Dictionary *scale* (the average number of translations available per query term).
- Dictionary *coverage rate*.
- The *crucial equivalent effect*.
- *Equivalent repetition* in dictionary entries and therefore in query translations.
- S-weighting of equivalents and terms according to ambiguity information found in the dictionary (The R-weight is the weight calculated for an equivalent by the retrieval engine, the S-weight any additional weight applied by the translation system which acts as a multiplier of the R-weight.

An S-weight applied before translation is known as a Q-weight, one applied after translation is a T-weight).

The bag of words queries extracted from our test set of 80 user requests constituted the test query set for our translation experiments (see chapter 3). In chapter 4, we established an upper bound on retrieval performance for our CLIR experiments by running a human-translated version of our test query set on the retrieval collection. We also obtained a so-called *Perfect Dictionary* translation, which we may view as an upper bound on dictionary-based CLIR experiments when queries are translated on a word-for-word basis. We need such an upper bound to determine the relative performance of any CLIR method. One cannot expect any set of query translations to perform better than human translations of the same query set. We obtained an upper bound of 33% AvP for the human translations and 31% for the *Perfect Dictionary* translations with R-weighting enabled.

7.2 Dictionary Scale

Dictionary *scale* is defined as as the average number of distinct translation equivalents listed in a given dictionary per query term. We performed Add-All-Equivalents translation of our test query set using a number of dictionaries derived from printed editions of bilingual English-French dictionaries, where scale ranged from 1.0 to around 6.0.

We discovered that query translations obtained using smaller-scale dictionaries performed better than those obtained with the use of larger-scale ones, provided a coverage level of 100% was maintained. Our largest-scale dictionary obtained absolute AvP of 13%, whereas our smallest-scale dictionary scored 25% for the same metric. There seemed to be a rough inverse correlation between dictionary scale and average overall performance. The improved performance of the smaller-scale dictionaries was due to a reduced *swamping effect* - the resulting translations contained fewer unwanted equivalents and so fewer documents matching these unwanted equivalents were present in retrieved document lists.

7.3 Dictionary Coverage Rate

The *coverage rate* of a dictionary is the percentage of query terms for which the system dictionary provides at least one equivalent. Due to the coverage needs of query translation, we opted to add entries from larger-scale dictionary to our smallest-scale dictionary in a process called *coverage compensation* to ensure 100% coverage for our experiments on scale.

Our experiments showed that a small-scale dictionary must provide coverage of at least the 20% most ambiguous terms in the query set prior to the application of coverage compensation in order to result in a performance improvement over a larger-scale dictionary alone. When implementing an equivalent selection strategy, one should concentrate one's effort on the 40% most ambiguous query terms, as reducing the number of equivalents provided for the other terms had no impact on performance. (Seeing how this last finding can be applied to the general case was beyond the scope of our experiments).

7.4 The Crucial Equivalent Effect

Not every query in our test set benefited from the use of a smaller scale dictionary for translation. This was due to what we called the *Crucial Equivalent Effect*. We defined the *Crucial Equivalent Effect* as the manner in which omitting a single equivalent from a query translation can result in radically lower performance for that query. We found that different translations of the same query were very sensitive to small differences in composition. This confirmed the conclusion reached by Hull and Grefenstette [47] that it is better to include many incorrect equivalents than to omit the most suitable for a given query term. The smaller the scale of the dictionary, the more we have to guard against this effect. We looked at a number of ways in which we could benefit from the reduced swamping effect of employing a smaller-scale dictionary for translation without suffering from a pronounced crucial equivalent effect.

Firstly, we combined a number of coverage compensated dictionaries of varying scale. By *combining* we mean concatenating the equivalent lists provided by each dictionary in the combination for each term and adding all of these equivalents to the query translation. We removed any S-weighting effects due to

equivalent repetition within a single term's total equivalent list. This approach was unsuccessful due to an increased swamping effect.

We then combined three similar coverage-compensated small-scale dictionaries of similar scale. Retrieval performance was similar using any of these three dictionaries alone. This strategy was as successful as using a single small-scale dictionary with coverage compensation once any S-weighting effects due to equivalent repetition within a single term's equivalent list were removed from the query translations, but without the massive drops in performance for individual queries due to the crucial equivalent effect - but was not significantly better.

Combining dictionaries in this way, therefore, did not solve the problems associated with the increased crucial equivalent effect.

7.5 Equivalent Repetition in Dictionary Entries within Query Translations

Sometimes a dictionary, particularly one of larger scale, will provide the same equivalent more than once for a given query term, for example, for different senses of the term. When Add-All-Equivalents translation is performing using a dictionary with entries of this type, these equivalents will be present more than once in the query translation. Each additional copy of a given equivalent present in a query translation increases its implicit default T-weight. Therefore, repeating equivalents is the same as applying additional T-weights to some equivalents and not to others.

We found that allowing equivalents to be repeated within single term's equivalent lists resulted in better retrieval performance than when equivalent repetition was removed before running the query translations on the retrieval collection. This was the case with both combinations of dictionaries tested. In fact, the less ambiguous equivalents tended to improve retrieval performance on their repetition, whereas the more ambiguous ones had the opposite effect. (This was a general trend noted only). IN addition, these results were observed in more than one set of experiments. Therefore, it would appear that equivalents should be S-weighted according to their level of ambiguity. (Ambiguity was calculated using the same dictionary across all experiments). We note that repetition within a query translation of an equivalent due to it being provided as a potential translation for more than one term was considered a separate phenomenon which was not investigated.

7.6 Additional S-Weighting Using Dictionary Information Only

In chapter 6, we looked at more formalised ways of increasing the importance of less ambiguous terms and equivalents in query translations and decreasing the influence of the more ambiguous ones. We applied a number of ad-hoc S-weighting strategies, using the *degree of ambiguity* of terms and equivalents as our basis. We defined the degree of ambiguity of a term as the number of distinct equivalents provided for it by the Collins-Robert Unabridged dictionary, and of an equivalent as the number of distinct translations listed for it in the English-French portion of the same dictionary.

A series of experiments demonstrated that deleting equivalents of a degree of ambiguity greater than 10 from the query translation improves performance. Furthermore, assigning uniform higher Q-weights to less ambiguous *query terms* prior to translation while assigning the default Q-weight of 1.0 to all other terms gave rise to significant improvements in retrieval performance. The higher the differential between the default Q-weight and the higher Q-weight assigned to less ambiguous terms, the lower the threshold value at which this strategy was still effective.

When assigning a uniform high Q-weight to less ambiguous terms, where two different values of Q-weight resulted in significantly better performance than the *CombinedNoRep* translations, there was no significant difference in performance between the two S-weighted translations. Finally, applying a stepped Q-weighting function to terms of degree of ambiguity lower than the threshold was not significantly better than assigning a uniform weight of 2.0 such to terms.

7.7 Conclusions

We combined all of the insights discussed in the above sections in one set of "best" query translations. Performance of 79% and 91% of the monolingual upper bound (French Human translations) AvP and R-P performance respectively, and 87% and 97% of the *Perfect Dictionary* AvP and R-P performance, was recorded (with R-weighting enabled). Comparison with the monolingual upper bound is the preferred measure employed in the field and our results compare well with the best reported in the literature. For example, Ballesteros and Croft [10] obtained 85% of the monolingual AvP with their co-occurrence statistics based strategy. This demonstrates that careful choice of dictionary according to characteristics and some simple S-weighting methods can be just as effective for CLIR as the more complex methods discussed in the literature, without the attendant costs and effort. **Therefore, this research has succeeded in helping to make developing a working CLIR system for a new language pair faster, cheaper and easier**.

7.8 Future Work

There were a number of avenues which, due to the time constraints of this project, could not be investigated further:

- Broadening of the electronic dictionaries used to encompass a large subset of English and French, as opposed to the 385-term "toy" dictionary employed here. This would allow the conclusions reached here to be verified on a wider request set.
- Testing of the current system using the request sets and document collections from the CLEF evaluations.
- Creation of many more CLIR dictionaries from printed sources, and also from other sources such as spell checkers, downloaded word lists etc, to verify the correlation observed between scale and performance.
- Investigation of the link between equivalent degree of ambiguity and frequency in the retrieval collection, to see if frequency information can be exploited to further improve performance.
- Investigating how the finding that one need only concentrate on the 40% more ambiguous bag of words query terms for equivalent selection generalises to a larger request set.
- Extending the system to a new language pair, specifically to a non-Indo-European language.
- Combining the methods described here with other CLIR methods.

These investigations would contribute further to making CLIR technology faster, easier and cheaper to implement.

Bibliography

[1] Altavista. http://www.altavista.com.

[2] Google web search engine. www.google.com.

[3] M. Adriani and W. B. Croft. The effectiveness of a dictionary-based technique for Indonesian-English cross-language text retrieval. Technical Report IR-170, University of Massachussetts, Amherst, 1997.

[4] M. Adriani and C. J. van Rijsbergen. Term similarity-based query expansion for cross-language information retrieval. In *Proceedings of the third European Conference on Research and Advanced Technology for Digital Libraries (ECDL '99)*, pages 311–322, 1999.

[5] M. Adriani and C. J. Van Rijsbergen. Phrase identification in cross-language information retrieval. In *Proceedings of the RIAO (Recherche d'Informations Assistée par Ordinateur) Conference: Content-Based Multimedia Information Access (RIAO 2000)*, pages 520–528, 2000.

[6] Beryl T. Atkins, Alain Duval, Rosemary C. Milne, Pierre-Henri Cousin, Hélène M. A. Lewis, Lorna A. Sinclair, Renée O. Birks, and Marie-Noëlle Lamy. *Collins Robert French-English English-Frnech Dictionary Unabridged*. Harper Collins, Glasgow, Scarborough, New York, third edition edition, 1993.

[7] J. P. Ballerini, M. Buchel, R. Domenig, D. Knaus, B. Mateev, E. Mittendort, P. Schauble, P. Sheridan, and M. Wechsler. SPIDER retrieval system at TREC-5. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-238: The fifth Text REtrieval Conference (TREC-5)*, pages 217–228, 1996.

[8] L. Ballesteros and W. B. Croft. Dictionary methods for cross-lingual information retrieval. In *Proceedings of the 7th International DEXA Conference on Database and Expert System Applications*, pages 791–801, 1996.

[9] L. 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 (SIGIR '97)*, pages 84–91, 1997.

[10] L. 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 (SIGIR '98)*, pages 64–71, 1998.

[11] Frédérique Bisson, Jérôme Charon, Christian Fluhr, and Dominique Schmit. EMIR at the CLIR track of TREC-7. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 337–342, 1998.

[12] M. Braschler, M.Y. Kan, and P. Schaüble. The Eurospider retrieval system and the TREC-8 cross-language track. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC-8)*, pages 367–376, 1999.

[13] M. Braschler, J. Krause, C. Peters, and P. Schäuble. Cross-language information retrieval (CLIR) track overview. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 25–32, 1998.

[14] M. Braschler, B. Mateev, E. Mittendorf, P. Schaüble, and M. Wechsler. SPIDER retrieval system at TREC-7. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 509–517, 1998.

[15] M. Braschler, B. Ripplinger, and P. Schaüble. Experiments with the Eurospier retrieval system for CLEF 2001. In *Workshop of Cross-Language Evaluation Forum, CLEF 2001, Working Notes*, 2001.

[16] M. Braschler and P. Schäuble. Multilingual information retrieval based on document alignment techniques. In *Proceedings of the 2nd European Conference on Researach and Advanced Technology for Digital Libraries (ECDL '98)*, pages 183–199, 1998.

[17] M. Braschler, P. Schäuble, and C. Peters. Cross-language information retrieval (CLIR) track overview. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC-8)*, pages 25–35, 1999.

[18] J. Broglio, J. P. Callan, and W. B. Croft. INQUERY system overview. In *Proceedings of the TIPSTER Text Program*, pages 47–67, 1994.

[19] P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer. The mathematics of statistical machine translation: parameter estimation. In *Computational Linguistics*, volume 19, pages 263–311, 1993.

[20] J. G. Carbonell, Y. Tang, R. E. Frederking, R. D. Brown, Y. Geng, and D. Lee. Translingual information retrieval: A comparative evaluation. In *Proceedings of the 15th International Joint Conference on Artficial Intelligence (IJCAI '97)*, pages 708–714, 1997.

[21] A. Chen, F. C. Gey, K. Kishida, H. Jiang, and Q. Liang. Comparing multiple methods for Japanese and Japanese-English text retrieval. In *Proceedings of the first NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition (NTCIR1)*, pages 49–58, 1999.

[22] A. Chen, H. Jiang, and F. Gey. English-Chinese cross-language IR using bilingual dictionaries. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-XXX: The ninth Text REtrieval Conference (TREC-9)*, 2001. To appear.

[23] H.-H. Chen, C.-C. Lin, and W.-C. Lin. Construction of a Chinese-English Wordnet and its application to CLIR. In *Proceedings of the 5th International Workshop on Information Retrieval with Asian Languages (IRAL 2000)*, pages 189–196, 2000.

[24] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, 1990.

[25] H. Cunningham, Y. Wilks, and R. Gaizauskas. GATE – a general architecture for text engineering. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 1057–1060, Copenhagen, 1996.

[26] M. W. Davis. New experiments in cross-language text retrieval at NMSU's computing research lab. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-238: The fifth Text REtrieval Conference (TREC-5)*, 1996.

[27] M. W. Davis and W. C. Ogden. Free resources and advanced alignment for cross-language text retrieval. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-240: The sixth Text REtrieval Conference (TREC-6)*, pages 385–402, 1997.

[28] M. W. Davis and W. C. Ogden. QUILT: Implementing a large-scale cross-language text retrieval system. In *Proceedings of the 20th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '97)*, pages 92–98, 1997.

[29] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic indexing. *Journal of the American Society for Information Science*, 41(6):1–13, 1990.

[30] A. Diekema, F. Oroumchian, P. Sheridan, and E. D. Liddy. TREC-7 evaluation of conceptual interlingua document retrieval (CINDOR) in English and French. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 169–180, 1998.

[31] C. Fellbaum, editor. *WordNet: an electronic lexical database*. MIT Press, 1998.

[32] C. Fluhr, D. Schmit, F. Elkateb, P. Ortet, and K. Gurtner. Multilingual database and crosslingual interrogation in a real internet application. In *Proceedings of the American Association for Artificial Intelligence Spring Symposium on Cross-Language Text and Speech Retrieval*, pages 32–36, 1997.

[33] M. Franz, J. S. McCarley, and S. Roukos. Ad hoc and multilingual information retrieval at IBM. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 157–168, 1998.

[34] A. Fujii and T. Ishikawa. Cross-language information retrieval at ULIS. In *Proceedings of the first NTCIR workshop on research in Japanese text retrieval and term recognition (NTCIR1)*, pages 163–169, 1999.

[35] P. Fung and L. Y. Yee. An IR Approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL '98)*, pages 414–420, 1998.

[36] F. Gey and A. Chen. TREC-9 cross-language information retrieval (English-Chinese) overview. In *NIST Special Publication 500-249: The ninth Text REtrieval Conference (TREC-9)*, pages 15–24, 2000.

[37] F. C. Gey, H. Jiang, A. Chen, and R. R. Laron. Manual queries and machine translation in cross-language retrieval and interactive retrieval with Cheshire II at TREC7. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 527–540, 1998.

[38] F. C. Gey, H. Jiang, V. Petras, and A. Chen. Cross-language retrieval for the CLEF collections - comparing multiple methods of retrieval. In *Workshop of Cross-Language Evaluation Forum, CLEF 2001, Working Notes*, 2001.

[39] L. Gravano. Merging ranks from hetereogenuous internet sources. In *Proceedings of the 23rd International Conference on Very Large Databases (VLDB '97)*, pages 196–205, 1997.

[40] D. Harman, editor. *The first Text REtrieval Conference (TREC-1)*. Department of Commerce, National Institute of Standards and Technology, November 1992.

[41] D. Harman. Overview of the fourth Text REtrieval Conference (TREC-4). In *NIST Special Publication 500-236: The fourth Text REtrieval Conference (TREC-4)*, 1995.

[42] D. K. Harman. Overview of the first Text REtrieval Conference (TREC-1). In D. K. Harman, editor, *NIST Special Publication 500-207: The first Text REtrieval Conference (TREC-1)*, pages 1–21, 1992.

[43] D. Hiemstra. Deriving a bilingual lexicon for cross-language information retrieval. In *Proceedings of the fourth Groningen International Information Technology Conference for Students*, pages 21–26, 1997.

[44] D. Hiemstra and F. de Jong. Cross-language information retrieval in Twenty-One: using one, some or all possible translations? In *Proceedings of the 14th Twente Workshop on Language Technology (TWLT14)*, pages 19–26, 1999.

[45] D. Hiemstra and W. Kraaij. Twenty-One at TREC-7: ad hoc and CLIR tracks. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 227–238, 1998.

[46] M. M. K. Hlava, G. Belonogov, B. Kuznetsov, and R. Hainebach. Cross language retrieval - English / Russian / French. In *AAAI 1997 American Association for Arfificial Intelligence Spring Symposium on Cross-Language Text and Speech Retrieval Series, Stanford University, California*, pages 63–83, 1997.

[47] D. A. Hull and G. Grefenstette. Experiments in multilingual information retrieval. In *Proceedings of the 19th ACM International Conference on Research and Development in Information Retrieval (SIGIR '96)*, pages 49–57, 1996.

[48] H. B. Hunt III, T. G. Szymanski, and J. D. Ullman. Operations on sparse relations and efficient algorithms for grammar problems. In *Proceedings of the 15th Annual IEEE Symposium on Switching and Automata Theory*, pages 127–132, 1974.

[49] G. Jones and A. M. Lam-Adesina. Exeter at CLEF 2001: Experiments with machine translation for bilingual retrieval. In *Workshop of Cross-Language Evaluation Forum, CLEF 2001, Working Notes*, 2001.

[50] G. Jones, T. Sakai, N. Collier, A. Kumano, and K. Sumita. Exploring the use of machine translation resources for English-Japanese cross-language information retrieval. In *Machine Translation Summit VII Workshop: Machine Translation for Cross Language Information Retrieval*, pages 15–22, 1999.

[51] G. Kikui. Term-list translation using monolingual word co-occurence vectors. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL '98)*, pages 670–674, 1998.

[52] G. Kikui. Resolving translation ambiguity using non-parallel bilingual corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL '99) workshop: Unsupervised Learning in Natural Language Processing*, 1999.

[53] J. L Klavans and E. Tzoukermann. *The encyclopaedia of artificial intelligence*, chapter Morphology. John Wiley and Sons, New York, 1987.

[54] W. Kraaij and D. Hiemstra. TREC-6 working notes: baseline tests for cross language retrieval with the Twenty-One system. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-240: The sixth Text REtrieval Conference (TREC-6)*, pages 753–760, 1997.

[55] W. Kraaij, R. Pohlmann, and D. Hiemstra. Twenty-One at TREC-8: Using language technology for information retrieval. In *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC-8)*, pages 285–300, 1999.

[56] K. L. Kwok. Evaluation of an English-Chinese cross-lingual retrieval experiment. In *Proceedings of the American Association for Artificial Intelligence Spring Symposium on Cross-Language Text and Speech Retrieval*, pages 133–137, 1997.

[57] K. L. Kwok. Exploiting a Chinese-English bilingual wordlist for English-Chinese cross language information retrieval. In *Proceedings of the 5th International Workshop on Information Retrieval with Asian Languages (IRAL 2000)*, pages 173–179, 2000.

[58] K. L. Kwok, L. Grunfeld, N. Dinst, and M. Chan. TREC-9 cross-language, web and question-answering track using PIRCS. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-XXX: The ninth Text REtrieval Conference (TREC-9)*, 2001. To appear.

[59] J. H. Lee. Combining multiple evidence from different relevance feedback methods. Technical Report IR-87, Centre for Intelligent Information Retrieval, University of Massachusetts, Amherst, 1996.

[60] G. A. Levow, D. W. Oard, P. Resnik, and C. I. Cabezas. Rapidly retargetable interactive translingual retrieval. In *Proceedings of the first International Conference on Human Language Technology Research (HLT 2001)*, pages 101–108, 2001.

[61] Ruiz M., A. Diekema, and P. Sheridan. CINDOR Conceptual interlingua document retrieval: TREC-8 evaluation. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC-8)*, pages 597–606, 1999.

[62] A. Maeda, F. Sadat, M. Yoshikawa, and S. Uemura. Query term disambiguation for web cross-language retrieval using a search engine. In *Proceedings of the 5th International Workshop on Information Retrieval with Asian Languages (IRAL 2000)*, pages 25–32, 2000.

[63] J. S. McCarley. Should we translate the documents or the queries in cross-language information retrieval? In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL '99)*, pages 208–214, 1999.

[64] P. McNamee, J. Mayfield, and C. Piatko. The HAIRCUT system at TREC-9. In *NIST Special Publication 500-249: The ninth Text REtrieval Conference (TREC-9)*, pages 273–294, 2001.

[65] T. Mori, T. Kokubu, and T. Tanaka. Cross-lingual information retrieval based on LSI with multiple word spaces. In *Proceedings of the second NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition (NTCIR2)*, pages 567–574, 2001.

[66] S. Nakazawa, T. Ochiai, K. Satoh, and A. Okumura. Cross language information retrieval based on comparable corpora. In *Proceedings of the first NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition (NTCIR1)*, pages 149–155, 1999.

[67] V. B. H. Nguyen, R. Wilkinson, and J. Zobel. Cross-language retrieval in English and Vietnamese. In *Proceedings of the American Association for Artificial Intelligence Spring Symposium on Cross-Language Text and Speech Retrieval, Stanford University, California*, 1997.

[68] J.-Y. Nie. TREC-7 CLIR using a probabilistic translation model. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 547–554, 1998.

[69] D. W. Oard. A comparative study of query and document translation for cross-language information retrieval. In *UMIACS Computational Linguistics Colloquium*, 1998.

[70] D. W. Oard. TREC-7 experiments at the University of Maryland. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-242: The seventh Text REtrieval Conference (TREC-7)*, pages 541–546, 1998.

[71] D. W. Oard and P. Hackett. Document translation for cross-language text retrieval at the University of Maryland. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-240: The sixth Text REtrieval Conference (TREC-6)*, pages 687–696, 1997.

[72] D. W. Oard, J. Wang, D. Lin, and I. Soboroff. TREC-8 experiments at Maryland: CLIR, QA and routing. In *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC-8)*, pages 623–636, 1999.

[73] Y. Ogawa. NTCIR advisor report. In *Proceedings of the first NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition (NTCIR1)*, pages 45–46, 1999.

[74] C. Peters, editor. *Cross-language information retrieval and evaluation: workshop of Cross-Language Evaluation Forum, CLEF 2000, revised papers*, number 2069 in Lecture Notes in Computer Science. Springer, 2000.

[75] A. Pirkola. The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '98)*, pages 55–63, 1998.

[76] M. F. Porter. An Algorithm for Suffix Stripping. *Program*, 14(3):130–137, July 1980.

[77] Prise retrieval system.
http://www.itl.nist.gov/iaui/894.02/works/papers/zp2/zp2.html.

[78] Y. Qu, A. N. Eilerman, J. Hongming, and D. A. Evans. The effect of pseudo-relevance feedback on MT-based CLIR. In *Proceedings of the RIAO (Recherche d'Informations Assistée par Ordinateur) Conference: Content-Based Multimedia Information Access (RIAO 2000)*, 2000.

[79] A.-M. Rassinoux, R. H. Baud, and J.-R. Scherrer. A multilingual analyser of medical texts. In *2nd International Conference on Conceptual Structures, ICCS '94, Berlin, Springer-Verlag*, pages 84–96, 1994.

[80] B. Rehder, M. L. Littman, S. Dumais, and T. K. Landauer. Automatic 3-language cross-language information retrieval with latent semantic indexing. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-240: The sixth Text REtrieval Conference (TREC-6)*, pages 233–240, 1997.

[81] S.E. Roberston. Overview of the Okapi projects. *Journal of Documentation*, 53(1):3–7, Jan 1997.

[82] T. Sakai. MT-based Japanese-English cross-language IR experiments using the TREC test collections. In *Proceedings of the fifth International Workshop on Information Retrieval with Asian Languages (IRAL 2000)*, pages 181–188, 2000.

[83] T. Sakai, S. E. Robertson, and S. Walker. Flexible pseudo-relevance feedback for NTCIR-2. In *Proceedings of the second NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition (NTCIR2)*, pages 559–566, 2001.

[84] T. Sakai, Y. Shibazaki, M. Suzuki, M. Kajiura, T. Manabe, and K. Sumita. Cross-language information retrieval for NTCIR at Toshiba. In *Proceedings of the first NTCIR workshop on research in Japanese text retrieval and term recognition (NTCIR1)*, pages 137–144, 1999.

[85] G. Salton. Experiments in multi-lingual information retrieval. *Information Processing Letters*, 2(1):6–11, 1973. also Technical Report TR 72-154 at Cornell University, 1972.

[86] G. Salton, E. Fox, and H. Wu. Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022, 1983.

[87] G. Salton and M. J. McGill. *Introduction to modern information retrieval.* McGraw-Hill Computer Science Series. McGraw-Hill, New York, 1983.

[88] J. Savoy. Report on CLEF-2001 experiments. In *Workshop of Cross-Language Evaluation Forum, CLEF 2001, Working Notes*, 2001.

[89] P. Schaüble and P. Sheridan. Cross-language information retrieval (CLIR) track overview. In *NIST Special Publication 500-240: The sixth Text REtrieval Conference (TREC-6)*, pages 623–636, 1997.

[90] P. Schaüble, P. Sheridan, and C. Peters. Cross-language information retrieval (CLIR) track overview. In *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC-8)*, pages 25–34, 1999.

[91] P. Sheridan and P Schäuble. Cross-language multi-media information retrieval. In *Third DELOS Workshop on Cross-Language Information Retrieval, number 97-W003 in ERCIM Workshop Proceedings*, 1997.

[92] M. Simard. Text-translation alignment: Aligning three or more versions of a text. In J. Véronis, editor, *Parallel Text Processing*, Text, Speech and Language Technology. Kluwer Academic Publishers, Dordrecht, 1999.

[93] Gilles Souvay. Categoriseur WinBrill - http://www.jupiter.inalf.cnrs.fr/WinBrill/. WWW Page, 1999. INALF, CNRS, France.

[94] K. Spärck Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval : development and status. Technical Report 446, Cambridge University Computer Laboratory, September 1998.

[95] Karen Spärck Jones. What is the role of NLP in text retrieval? In T. Strzalkowski, editor, *Natural Language Information Retrieval*. Kluwer, Dordrecht, 1999.

[96] Karen Spärck Jones and Peter Willett. *Readings in information retrieval*. The Morgan Kaufmann Series in Multimedia Information and Systems. Morgan Kaufmann, 1997.

[97] C. J. Van Rijsbergen. A theoretical basic for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33:106–109, 1977.

[98] C. J. Van Rijsbergen. *Information retrieval*. Butterworth, London, 2nd edition edition, 1979.

[99] E. M. Voorhees. Philosohpy of IR evaluation. In *Workshop of Cross-Language Evaluation Forum, CLEF 2001, Working Notes*, 2001.

[100] E. M. Voorhees and D. Harman. Overview of the eighth Text REtrieval Conference (TREC-8). In *NIST Special Publication 500-246: The eighth Text REtrieval Conference (TREC 8)*, pages 1–24, 1998.

[101] P. Vossen. EuroWordnet: a multilingual database for information retrieval. In *Third DELOS Workshop on Cross-Language Information Retrieval, Zurich*, 1997.

[102] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th International Conference on Research and Development in Information Retrieval (SIGIR '96)*, pages 4–11, 1996.

[103] J. Xu and R. Weischedel. TREC-9 cross-lingual retrieval at BBN. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-XXX: The ninth Text REtrieval Conference (TREC-9)*, 2001. To appear.

[104] P. G. Young. Cross-language information retrieval using latent semantic indexing. Master's thesis, Computer Science Department, University of Tennessee, Knoxville, 1994.

Appendix 3.(i)

The tables below display the English query set we obtained from the TREC CLIR topics from TREC-6, TREC-7 and TREC-8, and the corresponding
natural-language French queries obtained from the French topics from the same source. Note that we have removed stopwords and so on.

English	
QueryNum	**Query Contents**
1001	reason controversy surround waldheim world war ius action
1002	marriage increase worldwide
1003	measure stem international drug traffic
1004	possibility reusage garbage
1005	case study acupuncture
1006	air pollution automobile
1007	sex education combat aid
1008	rejection swiss referendum increase speed limit highway
1009	effect logging desertification
1010	information solar power car
1011	production organic cotton
1012	organic farming impact international trade
1013	attitude arab country peace process middle east
1014	effort combat international terrorism
1015	attitude law concern death penalty world
1016	reason resurgence tuberculosis world industrialize country
1017	article potato farming research consumption nutritional information
1018	perfume inflation proof luxury item world
1019	wine consumption production rise decrease world wide
1020	degree measure protect elephant affect world trade ivory
1021	measure fight grow child abuse world
1022	effect chocolate health
1023	successful spread american fast food franchise europe
1024	teddy bear gain popularity world wide
1025	demand foreign labor germany switzerland remain constant change fall berlin wall

English	
QueryNum	**Query Contents**
1026	bernese alpine railroad company longer communicate datum concern frequency traffic lotschberg
1027	worldwide oil pipeline united state oil pipe line country employ method deliver oil point origin ship point refinery safe pipeline environment economy
1028	reason destruction tropical forest south america consequence destruction
1029	arm force secret service hide truth disaster ustica
1030	famine sudan
1031	consequence german reunification
1032	involvement vatican failure banco ambrosiano
1033	latest development agricultural application genetic engineering
1034	economic social iron curtain raise
1035	import restriction trade barrier european community ec
1036	theft work art
1037	joint effort unity france west germany
1038	conversion debt poland
1039	measure propose national government european community ec favour integration immigrant
1040	british french cooperative effort development operation concorde supersonic jet
1041	difficulty involve military status unified germany
1042	economic cooperation united state airline european airline
1043	kidnapping end injury death
1044	extent issue freedom press prevent poland admit european council

English	
QueryNum	**Query Contents**
1045	election bosnium herzegovina
1046	swiss confederation public debt pay
1047	concern regard negative effect hole ozone layer public health justify
1048	development deployment high speed train
1049	case empirical investigation study concern environmental protection chemical plant field chemical industry
1050	main road accident
1051	consequence earthquake yunnan southwest china
1052	unemployment rate france
1053	problem raise unrestricted movement people europe
1054	dwindle supplies fish commercial fishery european community
1055	statistics legal illegal abortion world
1056	economic exploitation seabed continental shelf describe
1057	peace policy organisation african unity oau describe
1058	tourism italian adriatic coast decline considerably heavy concentration algae beach
1059	export low quality medicine switzerland world
1060	species rare bird illegally ship steal zoo profit
1061	discussion political decision deployment german arm force mission
1062	discovery location munitions remain world war ius
1063	artificial language rumantsch grischun manage ensure survival fourth swiss national language
1064	amount consequence chemical fertiliser agriculture discuss
1065	economic artistic situation european motion picture industry describe

English	
QueryNum	**Query Contents**
1066	european nation denounce military repression kurdish population turkish government suspend supply arm turkey
1067	danger manmade space debris pose
1068	legal rights homosexual individual couple
1069	leather industry market
1070	eta separatist movement active spain
1071	drag net fishing threaten survival dolphin death thousand animal ocean world
1072	variant means waste garbage disposal united state
1073	norm ilo international labour organisation industrial safety implementation describe
1074	procedure employ safe environmentally accept disposal nuclear waste
1075	earthquake sicily
1076	united state archeological site area yield information duration man presence united state
1077	country world euthanasia illegal common practise hospital
1078	development change economic policy slovenium
1079	reaction comment make slovenium dissolution parliament government kosovo
1080	proportion number communist european parliament increase
1081	world association individual commit save protect animal species

French	
QueryNum	**Query Contents**
2001	raison controverse egard agissement waldheim guerre mondial
2002	taux mariage augmenter monde
2003	mesure controler contrebande stupefiant
2004	possibilite recyclage ordure
2005	etude cas medical intervention acupuncture
2006	pollution atmosphere provoquee automobile
2007	introduction education sexuel ecole effort combattre escalage sida
2008	rejet initiative faveur augmentation vitesse autoroute suisse
2009	effet deforestation desertification
2010	information voiture solaire
2011	production usage coton ecologique
2012	agriculture ecologique influer commerce international
2013	attitude pays arabe egard processus paix moyen orient
2014	mesure combattre terrorisme international
2015	opinion legislation different pays concerner pein mort
2016	raison recrudescence tuberculose tiers monde pays industrialiser
2017	article culture pomme terre recherche consommation information nutritif pomme terre
2018	pourquoi parfum produit luxe affecter inflation
2019	consommation vin augmenter diminuer monde
2020	mesure protection elephant africain influence commerce international ivoire
2021	mesure combattre phenomene croissant maltraitance enfant monde
2022	chocolat effet quelconque sante
2023	succe croissance franchise fast food americain europe
2024	popularite ours peluche augmenter monde

French	
QueryNum	**Query Contents**
2025	demande main oeuvre etranger allemagne suisse changee restee pareil chute mur berlin
2026	pourquoi societe chemin fer alpin bernois bag communiquer chiffre indiquer frequence trafic travers lotschberg
2027	oleoduc mondial etat unir pays utiliser methode livrer petrole point origine point embarquement raffinerie mesure oleoduc representer menace environnement
2028	raison consequence destruction foret tropical amerique sud
2029	force armee service secret essayer cacher verite desastre ustica
2030	famine soudan
2031	consequence reunification allemand
2032	implication vatican faillite banco ambrosiano
2033	developpement application genie genetique agriculture
2034	voir creation nouveau rideau fer economique social
2035	limitation importation barriere tarifaire union europeen ue
2036	vol oeuvre art
2037	effort conjoindre unite france allemagne ouest
2038	conversion dette pologne
2039	mesure union europeen gouvernement pays union faveur integration immigrant
2040	cooperation franco anglaiser oeuvre utilisation avion supersonique concorde
2041	difficulte soulevee statut militaire allemagne unifiee
2042	cooperation economique compagnie aerien americain compagnie aerien europeen
2043	kidnapping terminer souvent blesse tue
2044	mesure question liberte presse empecher pologne admettre conseil europe

French	
QueryNum	**Query Contents**
2045	election bosnie herzegovine
2046	comment dette public suisse couvrir
2047	devoir inquieter effet trou couche ozone sante public
2048	construction service train haut vitesse
2049	recherche empirique etude traiter protection environnement site usine chimique domaine industrie chimique
2050	principal cause accident route
2051	consequence tremblement terre yunnan sud ouest chine
2052	taux chomage france
2053	probleme souleve libre circulation europe
2054	baisse stock poisson disposition poissonnerie commercial communaute europeen
2055	statistique concerner avortement legal illegal monde
2056	exploitation economique fond marin plateforme continental
2057	politique maintien paix organisation unite africain oua
2058	baisse tourisme cote adriatique italien devoir fort concentration algue autour plage
2059	exportation industrie pharmaceutique suisse medicament qualite suspect tiers monde
2060	espece oiseau rare vole zoo deloge illegalement tirer profit
2061	discussion decision politique concerner deploiement force armee allemand cadre mission onu
2062	decouverte localisation munition dater second guerre mondial
2063	rumantsch grischun assurer survie romanche
2064	quantite consequence utilisation engrais chimique agriculture
2065	situation economique artistique industrie europeen film

French	
QueryNum	**Query Contents**
2066	nation europeen fournir arme turquie
2067	danger representer debris spatial produire homme
2068	droit legal individu couple homosexuel
2069	comment porter marche cuir
2070	eta mouvement separatiste actif espagne
2071	peche filet mailler deriver serieux danger survie dauphin causer mourir millier mer ocean monde entier
2072	moyen nouveau derif traitement dechet etat unir
2073	norme protection professionnel oit organisation international travail oeuvre
2074	procedure utilisee envisagee traitement dechet nucleaire nuisible environnement
2075	cause tremblement terre sicile
2076	site zone archeologique etat unir fournir information duree presence humain etat unir
2077	cas euthanasie pratique illegalement hopital
2078	developpement changement politique economique slovenie
2079	ere reaction commentaire slovenie sujet dissolution parlement gouvernement kosovo
2080	proportion communiste parlement europeen progre
2081	association individu monde entier engager sauver espece protegee

Appendix 4.(i)

Chapter 5 - Control run experiments.

```
Unweighted - 12 58 66 69 81

Query 12 Run   Unweighted
-------------------------------------------------------
Human           7 Perfect       8
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
ecologique      8 influer       8
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
biologique      2 impact        8

Query 58
Run     Unweighted
-------------------------------------------------------
Human   33 Perfect              50
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
baisse  33 cote 50 devoir       50 autour       50
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
littoral        50 baisser      50
considerablement                50

Query 66
Run     Unweighted
-------------------------------------------------------
Human           17 Perfect      33
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
fournir         33
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
denoncer        25 militaire       33 repression   17
kurde           17 population      42 turc         17
gouvernement    17 interrompre     33
```

```
provisoirement  33 provision       33

Query 69
Run     Unweighted
-------------------------------------------------------
Human   11 Perfect      17
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
comment 22 porter       17
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
industrie               33

Query 81
Run     Unweighted
-------------------------------------------------------
Human            0   Perfect   17
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
individu        17 entier       8 engager 8
sauver           8 protege     25
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
proteger        8 animal        0 confier        17
individuel     17

Weighted - 3 5 14 43 48

Query 3
Run     Weighted
-------------------------------------------------------
Human           10 Perfect      16
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
mesure          20 controler    18
contrebande     18 stupefiant   28
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
demarche        17 arreter      23 international   13
drogue          13 trafic       13

Query 5
Run     Weighted
-------------------------------------------------------
Human   38 Perfect     100
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
medical 38 intervention 38
```

```
Query 14
Run     Weighted
-------------------------------------------------------
Human   12 Perfect      24
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
mesure  20 combattre    18
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
effort  16 lutter       35

Query 43
Run     Weighted
-------------------------------------------------------
Human   0 Perfect       50
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
souvent 50 blesse       50 tue     0
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
blessure                50 mort   50

Query 48
Run     Weighted
-------------------------------------------------------
Human           17 Perfect      40
-------------------------------------------------------
Adding to Perfect
-------------------------------------------------------
haut            17 construction 23 service 23
-------------------------------------------------------
Removing from Perfect
-------------------------------------------------------
developpement   47 utilisation  47 grand   20
```

Appendix 4.(ii)

Difference run details for the automatically-derived runs where the *AutoVerySmall* translations were best.

```
Queries - AutoVerySmall v. LargeNoRep - 13, 35, 48, 62

Query 13
--------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
LargeNoRep      7       11 AutoVerySmall   19      21
--------------------------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------------------------
aller           11      20 avancer         16      22
calme           16      20  campagne       16      20
citation        19      22 comparaitre     19      22
cortege         19      22 defiler         19      20
developper      20      25 direction       14      22
disposition     19      20 excroissance    19      22
lever           16      20 machine         19      20
methode         19      22 occuper         16      26
ordre           14      23 oriental        17      23
passer          13      20 patrie          16      20
position        17      25 proce           19      22
proceder        19      22 procedure       19      22
procession      19      22 protuberance    19      22
public          16      20 region          16      25
sommation       19      22 subir           19      22
taille          19      20 traitement      19      20
traiter         16      20 tranquillite    19      22
transformer     19      22 vers            14      22

Query 35
--------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
LargeNoRep      5       15 AutoVerySmall   25      35
--------------------------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------------------------
achat           25      35 affaire         15      30
alize           25      35 client          15      25
collectif       25      35 commercer       25      35
commercial      25      20 dire            20      25
echange         30      25 echanger        25      35
entretenir      15      30 groupement      20      30
```

```
importance      25      30 importer        20      25
limitation      15      30 metier          20      30
obstacle        25      25 portillon       25      35
propriete       25      25 relation        25      20
sens            20      30 signification   25
signifier       20      30 teneur          25      35
trafic          15      30 troquer         25      35
vouloir 20      25

Query 48
-------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
LargeNoRep      3       3  AutoVerySmall   17      17
-------------------------------------------------------------------------
Adding to AutoVerySmall
-------------------------------------------------------------------------
aigu            17      13 aller            7      17
altitude        10      13 amenagement     13      13
amphetamine     17      13 avance          13      13
braquer         17      13 brillier        17      17
caravane        17      13 citer            7      13
conduire        10      17 cortege         17      13
decocher        17      13 degre           13      13
derouelemnt     17      17 dresser         13      13
duree           10      17 eleve           13      13
elevee          17      13 entrainer       13      13
equipage        13      13 evolution       7       13
exceder         17      13 exercer         13      13
expansion       13      13 exploitation    13      13
exposer         13      13 exposition      13      13
faisander       17      17 ferme           13      13
file            13      13 formation       10      13
former          13      17 fort            3       20
grand           17      27 gros            7       13
haut2            3       3 industriel      13      17
instruire       17      13 intense         17      17
lancer          10      13 limitation      13      13
metro           17      13 noble           17      13
nouveau         17      20 obduration      17      17
paf             17      13 partir          3       20
preparer        13      17 profond         13      13
progres         10      13 promptitude     17      13
rame            23      33 rance           17      13
rapidite        17      17 recevoir        7       13
rougeaud        17      17 serie           10      13
speed           17      13 succession      17      13
suite            7      20 train2          13      20
trainee         17      13 trainer         17      17
travailler      17      20 usage           13      13
utilisation     10      13 valeur          10      13
vif             10      13 vite            13      13
vitesse2        17      30 zone            10      17
Query 62
-------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
LargeNoRep      0       8  AutoVerySmall   8       25
```

```
------------------------------------------------------------------------
Adding to AutoVerySmall
------------------------------------------------------------------------
activite        8       25 demeurer        8       25
eveil           8       25 exterieur       8       25
localisation    8       25 mondial         8       25
reperage        8       25 siecle          0       8
situation       0       17 trouvaille      8       8
univers         8       17 universel       8       25

Queries - AutoVerySmall v. AutoMediumNoRep - 20, 38, 50, 68

Query 20
------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
AutoMediumNoRep 11      33 AutoVerySmall   22      44
------------------------------------------------------------------------
Adding to AutoVerySmall
------------------------------------------------------------------------
affaire         22      44 affect          22      44
commercer       22      44 consequence     11      56
dose            22      44 elephanteau     22      44
entretenir      22      44 estimer         22      44
evaluer         22      44 mesurer         22      44
modifier        22      44 regle           22      44
sauvegarder     22      44 siecle          22      44
trafic          22      56

Query 38
------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
AutoMediumNoRep 14      11 AutoVerySmall   31      39
------------------------------------------------------------------------
Adding to AutoVerySmall
------------------------------------------------------------------------
creance         17      6  transformation  22

Query 50
------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
AutoMediumNoRep 3       1  AutoVerySmall   14      17
------------------------------------------------------------------------
Adding to AutoVerySmall
------------------------------------------------------------------------
canalisation    13      14 chemin          9       12
conduire        15      25 essentiel       13      18
evenement       13      17 fortuit         14      17
maitre          13      17 maitre          13      17
malheur         14      17 ocean           13      18
premier         5       22 rir             14      17

Query 68
------------------------------------------------------------------------
Run    Unweighted Weighted Run     Unweighted Weighted
AutoMediumNoRep 0       12 AutoVerySmall   18      35
------------------------------------------------------------------------
```

```
Adding to AutoVerySmall
------------------------------------------------------------------------
accoupler       18      35 associer        18      35
atteler         18      35 coupler         18      35
individu        18      35 legitime        12      35
original        18      35
```

Appendix 4.(iii)

Difference runs for the print-derived runs where *LargeTeensy* is better than the larger scale dictionaries.

```
queries 4 8 13 16 21 35 49  53 63

Teensy v. LargeNoRep
16 35 53

Query 16
--------------------------------------------------------------------------
Run    Unweighted Weighted        Unweighted Weighted
LargeNoRep      0       11 Teensy          22      44
--------------------------------------------------------------------------
Adding to Teensy
--------------------------------------------------------------------------
bon             0       33 calculer        22      33
campagne        22      33 cause           11      44
mondial         22      44 motif           22      44
patrie          22      44 raisonner       22      44
regain          22      44 region          0       33
sens            22      33 siecle          11      33
soutenir        0       33 univers         22      44
universel       22      33
--------------------------------------------------------------------------

Query 35
--------------------------------------------------------------------------
Run    Unweighted Weighted Unweighted Weighted
LargeNoRep      5       15 Teensy  25       35
--------------------------------------------------------------------------
Adding to Teensy
--------------------------------------------------------------------------
achat           25      35 affaire         15      30
alize           25      35 client          15      25
collectif       25      35 commercer       25      35
commercial      25      20 dire            20      25
echange         30      25 echanger        25      35
entretenir      15      30 groupement      20      30
importance      25      30 importer        20      25
limitation      15      30 metier          20      30
obstacle        25      25 portillon       25      35
propriete       25      25 relation        25      20
sens            20      30 signification   25      35
signifier       20      30 teneur          25      35
trafic          15      30 troquer         25      35
```

```
vouloir         20      25

Query 53
--------------------------------------------------------------------------
Run    Unweighted Weighted      Unweighted Weighted
LargeNoRep      0       0  Teensy         2       2
--------------------------------------------------------------------------
Adding to Teensy
--------------------------------------------------------------------------
activite        2       0 apparaitre      5       2
augmentation    2       0 augmenter       3       2
batir           2       0 caracteriel     2       2
construire      3       0 cultiver        2       2
difficile       3       3 edifier         2       2
elever          5       2 eriger          2       0
evoquer         3       2 famille         2       0
gens            2       0 geste           2       0
habitant        2       0 majorer         2       2
monter          2       2 nation          2       0
parent          2       2 peupler         2       0
population      3       0 poser           5       2
pousser         3       2 probleme2       2       3
procurer        3       2 provoquer       2       2
race            2       0 reculer         2       2
relance         2       0 relever         2       0
selles          2       2 soulever        3       2
superieur       2       0 these           2       0

Teensy v. SGemNoRep - 13 16 49

Query 13
--------------------------------------------------------------------------
Run    Unweighted Weighted Unweighted Weighted
SGemNoRep       3       6   Teensy  7       21
--------------------------------------------------------------------------
Adding to Teensy
--------------------------------------------------------------------------
calme           10      14 campagne        9       14
patrie          10      14 proceder        10      16
processus       9       26 region          7       17
taille          10      16 tranquillite    10      14
--------------------------------------------------------------------------
Removing from Teensy
--------------------------------------------------------------------------
procede         10      16

Query 49
--------------------------------------------------------------------------
Run    Unweighted Weighted        Unweighted Weighted
SGemNoRep        0       6         Teensy  11      14
--------------------------------------------------------------------------
Adding to Teensy
--------------------------------------------------------------------------
application     9       11 boiter          11      14
bureau          6       14 cas             11      14
chimique4       14      20 chimique3       11      17
```

```
concerner       9       14 domaine         9       17
entreprendre    6       14 etudier         9       17
etui            11      14 examiner        6       14
firme           6       14 inquietude      11      14
proce           11      14 produire2       6       14
produire1       11      14 souci           11      14
terrain         9       14 valise          11      14
zele            11      14
-------------------------------------------------------------------------
Removing from Teensy
-------------------------------------------------------------------------
environnement   9       9  examen          11      11
exercer         11      11 investigation   11      14
rapport         11      14

Teensy v. VerySmall - 8, 21,  63

Query 21
-------------------------------------------------------------------------
Run    Unweighted Weighted Unweighted Weighted
VerySmall       0       3  Teensy          6       9
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
abuser          6       9 bataille        6       9
battre          3       12 cultiver       6       9
devenir         3       9 insulte         6       9
mesurer         6       9 univers         6       9

Query 8
-------------------------------------------------------------------------
Run    Unweighted Weighted Unweighted Weighted
VerySmall       10      17  Teensy  12      20
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
agrandir        12      20  limiter 12      22


Query 63
-------------------------------------------------------------------------
Run    Unweighted Weighted Unweighted Weighted
VerySmall       46      85  Teensy  62      85
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
administrer     62      85 assurer         54      85
langage2        62      85 langue2         62      85
```

Appendix 4.(iv)

Difference runs where *LargeNoRep* and *AutoMediumNoRep* are better than *AutoVerySmall*.

```
Unweighted - AutoVerySmall v. LargeNoRep

Queries 34, 41, 74

Query 34

---------------------------------------------------------------------
Run     Unweighted
LargeNoRep      25 AutoVerySmall   0
---------------------------------------------------------------------
Adding to AutoVerySmall
---------------------------------------------------------------------
apparaitre      0        atel    0 augmentation    0   augmenter    0
baisser         0        batir   0 construire      25       coup    0
cultiver        0        donner  0 edifier         0        elever  0
eriger          0        etrier  0 evoquer         0        fer2    0
fete            0        garnir  0 lever2          0        majorer 0
mondain         0        monter  0 pousser         0       procurer 0
provoquer       0        rappel  0  rapporter      0        reculer 0
relance         0        relever 0 rentable        0       repasser 0
rideau4         0        rideau3 0 rideau2         0       sociable 0
social          0       soulever 0 superieur       0        voile   0

Query 41
---------------------------------------------------------------------
Run     Unweighted
LargeNoRep      23 AutoVerySmall   13
---------------------------------------------------------------------
Adding to AutoVerySmall
---------------------------------------------------------------------
armee            2 entrainer       12 meler           13
necessiter      13 obstacle        19 position        10
prestige        13 standing        13 statut          33

Query 74
---------------------------------------------------------------------
Run     Unweighted
LargeNoRep      4 AutoVerySmall   0
---------------------------------------------------------------------
Adding to AutoVerySmall
---------------------------------------------------------------------
```

```
abandon         0 admettre        0 arrangement     0 atomique        0
cession         0 charger         0 coffre          0 danger2         0
dechet          0 desamorcage     0 desert          0 desolee         0
desoler         0 destruction     0 disposition     0 eau2            4
eau1            0 exeuction       0 expedition      0 fort            0
friche          0 gachis          0 garder          0 gaspiller       0
inculte         0 inutilise       0 laisser         0 livraison       0
manger          0 ordure          0 passer          0 perdre          0
pertprudent     0 raisonable      0 recourir        0 resigner        0
resolution      0 risque          0 sale2           4 sale1           4
securite        0 solide          4 soumettre       0 superflu        0
supprimer       0 terrain         0 terre           0 usage           0
use             0 usee            0 utiliser        0 vague           0
vente           0 zigouiller      0
```

```
Unweighted - AutoVerySmall v. AutoMediumNoRep


Queries 4, 31, 40

Query 4
-----------------------------------------------------------------------
Run     Unweighted
AutoMediumNoRep 13 AutoVerySmall   4
-----------------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------------
dechet  11 detritus        7 eventualite     4

Query 31
-----------------------------------------------------------------------
Run     Unweighted
AutoMediumNoRep 41 AutoVerySmall   39
-----------------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------------
allemagne       32 importance      43 suite   32

Query 40
-----------------------------------------------------------------------
Run     Unweighted
AutoMediumNoRep 50 AutoVerySmall   30
-----------------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------------
anglais 30         avion     60  cooperative  30 exposer 30
fonctionnement  30 formation 30 france        20 giclee  30
gicler  30         jaillir   30 jais          30 jicler  30
voyager 30


Weighted - AutoVerySmall v. LargeNoRep

Queries 12, 14, 41
```

```
Query 12
-----------------------------------------------------------------
Run     Weighted
LargeNoRep      10
AutoVerySmall   2
-----------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------
achat           0 affaire         0 alize           2 biologique     30
chimique        3 choc            0 client          2 commercer       0
commercial      0 cultiver        3 echange         2 echanger        3
effet           2 enfoncer        2 engrais        12 entretenir      0
fondemental     2 influer         2 insecticide     2 metier          2
naturel         0 ni              0 presser         2 relation        0
trafic          0 troquer         2

Query 14
-----------------------------------------------------------------
Run     Weighted
LargeNoRep      20 AutoVerySmall   16
-----------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------
combattre       18 lutter  23

Query 41
-----------------------------------------------------------------
Run     Weighted
LargeNoRep      21 AutoVerySmall   12
-----------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------
armee           12 entrainer        8
meler           12 necessiter      11
obstacle        15 position        23
standing        12 statut  36


Weighted - AutoVerySmall v. AutoMediumNoRep
queries - 43, 66, 71

Query 43
-----------------------------------------------------------------
Run     Weighted
AutoMediumNoRep             25 AutoVerySmall             0
-----------------------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------------------
achever    0 dece            0 effondrement 0 extremite   0
finir      0 kidnapping      50 lesion      0 queue       0
rapt       0 terminer        0  tort        0

Query 66

-----------------------------------------------------------------
Run     Weighted
```

```
AutoMediumNoRep         25 AutoVerySmall 8
------------------------------------------------------------------------
Adding to AutoVerySmall
------------------------------------------------------------------------
arme     33 armee        8 armer       17 branche    8
dinde     8  dindon      8 direction    8 fournir    8
gestion   8  munir       8 ogive        8 peuple    25
procurer  8 refoulement 17 reserve      8 souplement 8
soupless  8 stock        8 surseoir     8 tete       8

Query 71
------------------------------------------------------------------------
Run     Weighted
AutoMediumNoRep            33 AutoVerySmall   0
------------------------------------------------------------------------
Adding to AutoVerySmall
------------------------------------------------------------------------
araigne  0 arriere 0 brute   0 chasser 0
dece     0 effondrement      0 entrainer 0
filet3  50 filet2           33 filet1  33
gagner   0 mille             0 net4    0
net2     0 net1              0 poser   0
produire 0 rapporter         0 rester  0
siecle   0 survivance        0 tendre  0
tirer    0 traineau          0 trainer 0
tulle    0 vestige           0 voile   0
```

Appendix 4.(v)

Difference runs for the print-derived dictionaries where the larger scale dictionaries are better than *Teensy*.

```
LargeNoRep v.Teensy 31 51 74

Query 31
-------------------------------------------------------------------------
Run    Unweighted Weighted     Unweighted Weighted
LargeNoRep      41      50      Teensy  32      45
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
importance      41      50
suite           36      48

Query 51
-------------------------------------------------------------------------
Run     Unweighted      Weighted
LargeNoRep      75      83
Teensy          67      67
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
chine           67      67 importance      8       67
procelain       8       67 seisme         67      75
suite           17      67


Query 74
-------------------------------------------------------------------------
Run     Unweighted      Weighted  Unweighted      Weighted
LargeNoRep      4       4 Teensy          0       0
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
abandon         0       0 admettre        0       0
arrangement     4       4 cession         0       0
charger         0       0 coffre          0       0
danger          0       0 danger          0       0
dechet          0       8 desamorcage     0       0
desert          4       4 desolee         0       0
desoler         0       0 destruction     0       0
eau2            0       0 eau1            0       0
enlevement      0       0 exeuction       0       0
expedition      0       0 fort            0       0
```

```
friche          0       0 gachis          0       0
garder          0       0 gaspillage      0       0
gaspiller       0       0 hors            0       0
inculte         0       0 inutilise       0       0
laisser         0       0 livraison       0       4
manger          0       0 ordure          0       0
passer          0       0 perdre          0       0
perte           0       0 prudent         0       0
raisonable      0       0 recourir        0       0
resigner        0       0 resolution      0       0
risque          0       0 sale2           4       4
sale1           0       0 securite        0       0
solide          0       0 soumettre       0       0
superflu        0       0 supprimer       0       0
terrain         0       4 terre           0       0
usage           0       0 use             0       0
usee            0       0 utiliser        0       0
vague           0       0 vente           0       0
zigouiller      0       0
-------------------------------------------------------------------------

SGemNoRep v Teensy 12 32 33

Query 12
-------------------------------------------------------------------------
Run     Unweighted      Weighted Unweighted      Weighted
SGemNoRep       13      23              Teensy  10      20
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
metier  10      20
-------------------------------------------------------------------------
Removing from Teensy
-------------------------------------------------------------------------
effet   13      22 enfoncer        10      20
influer 10      22 presser         10      22

Query 32
-------------------------------------------------------------------------
Run     Unweighted      Weighted Unweighted      Weighted
SGemNoRep       31      88       Teensy  4       85
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
echec   4       77
-------------------------------------------------------------------------
Removing from Teensy
-------------------------------------------------------------------------
complication    4       88 difficulte      8       88
insucces        4       85 probleme        35      88

Query 33
-------------------------------------------------------------------------
Run     Unweighted      Weighted Unweighted      Weighted
SGemNoRep       14      43       Teensy  0       29
-------------------------------------------------------------------------
```

```
Removing from Teensy
-------------------------------------------------------------------------
genique          0       29 machination   0       29

Query 10

-------------------------------------------------------------------------
Run     Unweighted      Weighted Unweighted      Weighted
VerySmall       15      25       Teensy  7       21
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
voiture          15      25

Query 59
-------------------------------------------------------------------------
Run     Unweighted      Weighted Unweighted      Weighted
VerySmall       17      17       Teensy  0       0
-------------------------------------------------------------------------
Adding to Teensy
-------------------------------------------------------------------------
exporter         17      17       univers 0       0
vil               0       0
```

Appendix 4(vi)

Difference runs between *AutoVerySmall* translations and translations formed by combining all 3 auto-derived dictionaries, where the *CombinedAuto* translations performed better.

```
Unweighted - 4 6 9 41 55

Query 4
Run     Unweighted
-------------------------------------------------------
AutoVerySmall            4 CombinedAuto 13
-------------------------------------------------------
Adding to AutoVerySmall
-------------------------------------------------------
dechet2  11 dechet1     11 detritus2    9
detritus1 7 eventualite2 0 eventualite1 4
foutaise1 4 ordure3      7 ordure2      4
parasite1 4 possibilite3 4 possibilite2 4
rebut1    4 reusage3     4 reusage2     4

Query 6
Run Unweighted
-------------------------------------------------------
AutoVerySmall  37 CombinedAuto   38
-------------------------------------------------------
Adding to AutoVerySmall
-------------------------------------------------------
aerer2         33 aerer1         36 air3        31
air2           31 aspect1        33 auto2       29
auto1          37 automobile3    39 automobile2 39
brise1         34 connaitre2     14 connaitre1  33
contamination2 29 contamination1 34 diffuser1   31
exhaler2       37 exhaler1       37 leger1      31
mine1          33 pollution3     36 pollution2  36
profanation2   33 profanation1   37 souffle1    36
tapis1         36


Query 9
Run Unweighted
-------------------------------------------------------
AutoVerySmall     2 CombinedAuto      8
-------------------------------------------------------
Adding to AutoVerySmall
-------------------------------------------------------
action2           4 action1           4 amener1      2
```

```
apporter2         4 apporter1         6 bois2       10
bois1            10 consequence2      4 consequence1 6
desertification3 25 desertification2 10 effectuer2   0
effectuer1        2 effet3            2 effet2       2
exploitation3     2 exploitation2     2 obtenir2     2
obtenir1          2 operer1           2 realiser1    4
sens1             2

Query 41
Run Unweighted
--------------------------------------------------------
AutoVerySmall            13 CombinedAuto 22
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
allemagne3 13 allemagne2 13 armee2       2
armee1      2 difficulte3 6 difficulte2  6
entrainer2  2 entrainer1 12 impliquer3   2
impliquer2  2 meler2     13 meler1      13
militaire3 12 militaire2 12 necessiter1 13
obstacle2   9 obstacle1  19 position2   22
position1  10 prestige1  13 situation3   7
situation2  7 standing1  13 statut2     35
statut1    33 unifie3    13 unifie2     13

Query 55
Run Unweighted
--------------------------------------------------------
AutoVerySmall 22 CombinedAuto 44
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
avortement3   44 avortement2  44 chiffre2      22
chiffre1      22 grossesse2   56 grossesse1    56
illegal3      11 illegal2     11 interruption2 44
interruption1 56 judiciaire1  11 juridique1    11
legal3        22 legal2       22 legal1        22
legitime2     11 legitime1    22 mensuration2  22
mensuration1  22 monde3       22 monde2        22
mondial1      22 siecle2      11 siecle1       22
statistique3  33 statistique2 44 univers1      22
universel1    22 volontaire2  22 volontaire1   33
--------------------------------------------------------

weighted - 11 14 43 52 71

Query 11
Run Weighted
--------------------------------------------------------
AutoVerySmall 0 CombinedAuto 14
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
biologique1   0 chimique1    29 coton3       0
coton2       14 cultiver1     0 engrais1    14
fabrication2 14 fabrication1 14 fil2         0
```

```
fil1          0 fondemental2  0 fondemental1 0
insecticide1 14 naturel2      0 naturel1     0
ni1           0 oeuvre1       0 ond1         0
organique3    0 organique2    0 piece1       0
production3  14 production2  14 realisation2 0
scene1        0

Query 14
Run Weighted
--------------------------------------------------------
AutoVerySmall  16 CombinedAuto   22
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
combat3         2 combat2         4 combattre2   6
combattre1     18 effort3        16 effort2     16
international3 29 international2 22 lutter2     12
lutter1        22 terrorisme3    27 terrorisme2 22

Query 43
Run Weighted
--------------------------------------------------------
AutoVerySmall    0 CombinedAuto 50
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
aboutissement1   0 accomplir1   0 achevement1   0
achever2        50 achever1     0 ailier1       0
aneantissement1  0 arriver1     0 atteindre1    0
blessure3        0 blessure2    0 bout3         0
bout2            0 but1         0 cesser1       0
conclure1        0 dece2        0 dece1         0
dessein1         0 echeance1    0 effondrement2 0
effondrement1    0 enlevement3  0 enlevement2   0
expirer1         0 extremite2   0 extremite1    0
finir2           0 finir1       0 issue1        0
kidnapping2     50 kidnapping1 50 lesion2       0
lesion1          0 limite1      0 mort3         0
mort2            0 prejudice1   0 queue2        0
queue1           0 rapt2        0 rapt1         0
restant1         0 rester1      0 terminer2     0
terminer1        0 tort2        0 tort1         0

Query 52
Run Weighted
--------------------------------------------------------
AutoVerySmall 4 CombinedAuto 16
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
allure1     4 chomage3    40 chomage2    32
classe2     4 classe1      4 classer2     4
classer1    4 considerer2 12 considerer1 12
cours1      8 evaluer2     8 evaluer1     8
fixer2      8 fixer1       8 france3     44
france2    56 loyer2       0 loyer1       4
```

```
matriciel2  4 matriciel1   4 meriter1     4
proportion3 4 proportion2  4 tarif1       4
taux2      36 taux1       44 train1       4
vitesse2    0 vitesse1     4

Query 71
Run Weighted
-----------------------------------------------------
AutoVerySmall   0 CombinedAuto 33
-----------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------
aneantissement1 0 animal3       0 animal2    0
araigne2        0 araigne1      0 arriere2   0
arriere1        0 bouffee1      0 boulet1    0
brute2          0 brute1        0 casser1    0
chasser2        0 chasser1      0 corvee1    0
dauphin3       17 dauphin2     17 dece2      0
dece1           0 drag1         0 drague3    0
drague2         0 draguer1      0 drege1     0
effondrement2   0 effondrement1 0 entrainer2 0
entrainer1      0 entrave1      0 filer3     0
filer2          0 filet6       67 filet5    50
filet4         50 filet3       50 filet2    33
filet1         33 frein2        0 frein1     0
frotter1        0 gagner2       0 gagner1    0
gripper1        0 hers1         0 languir1   0
meler1          0 menacer3      0 menacer2   0
mille3          0 mille2        0 monde3     0
monde2          0 mondial1      0 mort3      0
mort2           0 net8          0 net7       0
net6            0 net5          0 net4       0
net3            0 net2          0 net1       0
ocean3         17 ocean2       17 patin1     0
pecher3        17 pecher2      17 pied1      0
piston1         0 poser2        0 poser1     0
produire2       0 produire1     0 rafle1     0
rapporter2      0 rapporter1    0 raseur1    0
raseuse1        0 resistance1   0 rester2    0
rester1         0 sabot1        0 seine1     0
siecle2         0 siecle1       0 survie3    0
survie2         0 survivance2   0 survivance1 0
tendre2         0 tendre1       0    tiras1  0
tirer2          0 tirer1        0 toucher1   0
traineau2       0 traineau1     0 trainee1   0
trainer2        0 trainer1      0 travesti1  0
tulle2          0 tulle1        0 univers1   0
universel1      0 vestige2      0 vestige1   0
voile2          0 voile1        0

Query 13
Run Unweighted
-----------------------------------------------------
AutoVerySmall 19  CombinedAuto 10
-----------------------------------------------------
Adding to AutoVerySmall
```

```
-----------------------------------------------------
aller1         11 arabe3         19 arabe2        19
attitude3      11 attitude2      13 avancer2      10
avancer1       16 calme1         16 campagne2     11
campagne1      16 citation1      19 comparaitre1  19
cortege2       19 cortege1       19 defiler2      19
defiler1       19 developper1    20 direction2     9
direction1     14 disposition2   11 disposition1  19
excroissance1  19 lever2         11 lever1        16
machine1       19 methode1       19 milieu3        7
milieu2        11 ordre1         14 orient3       19
orient2        19 oriental2      13 oriental1     17
paix3          19 paix2          19 passer1       13
patrie2        13 patrie1        16 pays3         16
pays2          17 position2      19 position1     17
proce1         19 proceder2      11 proceder1     19
procedure2     13 procedure1     19 procession1   19
processus3     19 processus2     16 public1       16
region1        16 sommation1     19 subir2        11
subir1         19 taille2        16 taille1       19
traitement1    19 traiter2       11 traiter1      16
tranquillite2  19 tranquillite1  19 transformer2  11
transformer1   19 vers2           7 vers1         14

Query 19
Run Unweighted
-----------------------------------------------------
AutoVerySmall 17 CombinedAuto 0
-----------------------------------------------------
Adding to AutoVerySmall
-----------------------------------------------------
affaiblir1        17 affaiblissement1 17 aigu1          17
aller1            25 amoindrir2       17 amoindrir1     17
amoindrissement2  17 amoindrissement1 17 ample2         17
ample1            17 augmentation1    17 augmenter1     33
baisser1          17 but2              0 but1           17
calmer1           17 clore1           17 considerable1  25
consommation3     25 consommation2    25 consomption2   17
consomption1      17 cote1            25 croire1        17
croitre1          17 debout2          17 debout1        17
decroissance2     17 decroissance1    17 decroitre2     17
decroitre1        17 devenir2          0 devenir1       17
diminuer3         25 diminuer2        25 diminution2    25
diminution1       33 dresser1         17 elevation2     17
elevation1        17 elever1          17 eminence1      17
envergure1        17 etendre1         17 fabrication2    0
fabrication1      17 flot1            17 flotter1       17
flux1             17 grand2            0 grand1          0
grandir1          17 hausse2           0 hausse1        17
hauteur1          17 houleux1         17 immense1       17
large3             0 large2            8 lever4          0
lever3             0 lever2           17 loin2           0
loin1             17 majoration1      17 monde3          8
monde2             8 mondial1          8 monter1        17
naissance1        17 oeuvre1          17 ond1          17
origine1          25 pente1           17 phtisie2      17
```

```
phtisie1         17 piece1           17 presentation1 17
production3      17 production2      17 realisation1  17
reduire2          0 reduire1         17 refroidir1    17
refroidissement1 17 relevement1      17 relever2       8
relever1         25 remonter1        17 revolter1     17
scene1           17 seance1          17 session1      17
siecle1          17 soulever1        17 source1       17
univers1         17 universel1       17 vaste1        17
vin3             33 vin2             33

Query 63
Run Unweighted
--------------------------------------------------------
AutoVerySmall  54 CombinedAuto 38
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
administrer2   54 administrer1 54 affecter1      54
artificiel3    46 artificiel2  54 assurer3       23
assurer2       39 debrouiller2 54 debrouiller1   54
diriger2       46 diriger1     62 echelon2       54
echelon1       54 ecouter1     54 ensemble1      54
etudier1       54 exploiter1   54 factice1       54
feindre1       54 fors1        54 garantir2      31
garantir1      46 gerer3       31 gerer2         39
grand2         31 grand1       39 grischun3      70
grischun2      70 journal1     54 langage4       31
langage3       46 langage2     54 langage1       54
langue6        54 langue5      54 langue4        54
langue3        54 manier1      54 manoeuvrer1    54
mener1         54 national3    46 national2      54
pays1          54 quart2       54 quart1         54
quatrieme3     39 quatrieme2   46 ressortissant2 46
ressortissant1 54 rumantsch3   70 rumantsch2     70
savoir1        46 suisse3      46 suisse2        46
survie3        31 survie2      46 survivance2    54
survivance1    54 synthetique2 54 synthetique1   54
truquer1       54 vestige2     54 vestige1       54

Query 67
Run Unweighted
--------------------------------------------------------
AutoVerySmall 38 CombinedAuto 31
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
affectation1  38 air2          8 air1          8
artificiel2   15 artificiel1 46 attitude1     8
blanc1        31 creer1        8 danger3      38
danger2       38 debris3      46 debris2      38
detritique2   38 detritique1 38 donner2      15
donner1       23 echelonner2 15 echelonner1 38
espace3       38 espace2      38 espacer2     38
espacer1      38 formuler1    15 frimer2      38
frimer1       38 interligne1 38 laps1        38
peril2         8 peril1       38 periode1      8
```

```
place2         0 place1       8 pos1        38
pose2          8 pose1       38 poser3       8
poser2         8 presenter1  38 roche2       8
roche1        38 synthetique3 8 synthetique2 8

Weighted - 16 38 56 57 69

Query 16
Run Weighted
--------------------------------------------------------
AutoVerySmall   44 CombinedAuto    33
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
bon1            33 calculer2        0 calculer1    33
campagne2       33 campagne1       33 cause1       44
industrialiser3 22 industrialiser2 22 monde3       33
monde2          44 mondial1        44 motif2       33
motif1          44 patrie2         33 patrie1      44
pays3           33 pays2           44 raison3      33
raison2         33 raisonner2      11 raisonner1   44
reapparition3    0 reapparition2   11 redemarrage2 11
redemarrage1    44 region1         33 sens1        33
siecle2         11 siecle1         33 soutenir2    33
soutenir1       33 tuberculose3    56 tuberculose2 44
univers1        44 universel1      33

Query 38
Run Weighted
--------------------------------------------------------
AutoVerySmall   39  CombinedAuto   19
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
conversion3      6 conversion2      6 dette3 50
dette2          55 pologne3        83 pologne2 89
transformation2 11 transformation1 28


Query 56
Run Weighted
--------------------------------------------------------
AutoVerySmall 20  CombinedAuto 18
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
continental3  14 continental2  14 decrire3      0
decrire2      12 depeindre2    10 depeindre1   18
description2  10 description1  18 eceuil1      20
economique3   18 economique2   18 etagere3     20
etagere2      20 europeen2     16 europeen1    20
exploitation3 24 exploitation2 29 fond3         6
fond2         18 mer3          27 mer2         29
place1        18 planche2      10 planche1     18
qualifier1    20 rapporter2    16 rapporter1   20
rayon2        10 rayon1        18 rebord1      20
```

```
rentable2      8 rentable1     18 representer1 18
saillir1      20

Query 57
Run Weighted
--------------------------------------------------------
AutoVerySmall 65 CombinedAuto  50
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
accord2       55 accord1       65 africain3    53
africain2     63 association1  65 cadre1       58
calme1        65 decrire3      33 decrire2     65
depeindre2    58 depeindre1    65 description2 63
description1  65 harmonie2     60 harmonie1    65
ligne2        65 ligne1        65 ordre1       65
organisation3 63 organisation2 65 organisme2   65
organisme1    65 oua3          70 oua2         68
paix3         63 paix2         60 police1      63
politique3    45 politique2    53 public1      65
qualifier1    63 regle1        65 representer1 63
tranquillite2 60 tranquillite1 65 unite3       65
unite2        65

Query 69
Run Weighted
--------------------------------------------------------
AutoVerySmall 67 CombinedAuto 28
--------------------------------------------------------
Adding to AutoVerySmall
--------------------------------------------------------
aller2        39 aller1     77 application1 61
assiduite2    56 assiduite1 67 chamois2      0
chamois1      61 clientele1 39 commission2  17
commission1   67 cuir3      67 cuir2        67
deboucher2     0 deboucher1 50 industrie3   61
industrie2    61 lancer2    22 lancer1      67
larfeuil2     67 larfeuil1  67 marche3      33
marche2       67 peau2       0 peau1        56
portefeuille1 33 trouver1   67 vendre2      11
vendre1       50 zele2       0 zele1        61
```

Appendix 5.(i)

Comparing the highest scoring dictionary's translation with the next highest.

```
Queries where SGemNoRep was best.

Query 48
--------------------------------------------------------
Run Result
--------------------------------------------------------
SGemNoRep 23 InsightNoRep      0
--------------------------------------------------------
drogue    23 file 23 filer    23 formation 23
instruire 23 ivre 23 recevoir 23
vite      23 rame 7 nouveau   23
grand     23


Query 52
--------------------------------------------------------
Run Result
--------------------------------------------------------
SGemNoRep 32  InsightNoRep 12
--------------------------------------------------------
classer   36 evaluer 40

Query 55
--------------------------------------------------------
Run Result
--------------------------------------------------------
SGemNoRep 56 InsightNoRep 22
--------------------------------------------------------
juridique 44 statistiques 56 statistique 33

Queries where SLangNoRep was best.

Query 10
--------------------------------------------------------
Run Result
--------------------------------------------------------
SLangNoRep 24  SGemNoRep   10
--------------------------------------------------------
auto       21 connaissance 22 wagon    21
actionner  24 automobile   22 autorite 25
capacite   25 energie      19 marcher  22
pouvoir    24 propulser    22 soleil   28
```

```
Query 13
-------------------------------------------------------
Run Result
-------------------------------------------------------
SLangNoRep 23  SGemNoRep 6
-------------------------------------------------------
calme      23 region    23 taille 23
vers       19 millieu   23 moyen  19
orient     17

Query 59
-------------------------------------------------------
Run Result
-------------------------------------------------------
SLangNoRep 33 SGemNoRep 17
-------------------------------------------------------
deprimer   33 inferieur 17
mauvais    17 monde2    17

Queries where InsightNoRep was best.

Query 17
-------------------------------------------------------
Run Result
-------------------------------------------------------
InsightNoRep 49  SGemNoRep 43
-------------------------------------------------------
connaissance (no need to run)

Query 50
-------------------------------------------------------
Run Result
-------------------------------------------------------
InsightNoRep 15 SGemNoRep 1
-------------------------------------------------------
conduite     14  hasard  15
rir          15  hazard  15

Query 58
-------------------------------------------------------
Run Result
-------------------------------------------------------
InsightNoRep 83 SGemNoRep 50
-------------------------------------------------------
baisser      83 charge    67 echouer 83
rouler       83 violent   83
```

Appendix 5.(ii)

Comparison of *Combined* and *CombinedNoRep* translations, with retrieval engine R-weighting enabled and disabled.

```
Queries where Combined was best, unweighted.

Query 8
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined     12 CombinedNoRep 5
-------------------------------------------------------
augmentation3 3 augmentation2 5 augmenter3  7
augmenter2   10 filer2        5  limite3   12
limite2      12 limiter3      0 limiter2    3
national3     3  national2    3 principal2  7
rapidite3     3 rapidite2     3 referendum3 0
referendum2   3 rejet3        7 rejet2      7
route5       15 route4       15 route3     15
route2       15 suisse4       5 suisse3     5
suisse2       5 vite2         3 vitesse3   15
vitesse2     15

Query 14
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined       2 CombinedNoRep  0
-------------------------------------------------------
combat3        2 combat2        2 combattre3   2
combattre2     2 effort3        2 effort2      0
international3 0 international2 0 terrorisme3 20
terrorisme2   16

Query 68
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined    6CombinedNoRep 0
-------------------------------------------------------
couple3     6 couple2      6 droit3       0
droit2      0 homosexuel3 35 homosexuel2 18
individu3   6 individu2    6 individuel3  0
individuel2 0 juridique2   0 legal3       0
legal2 0
```

```
Query 69
--------------------------------------------------------
Run Result
--------------------------------------------------------
Combined       17 CombinedNoRep  11
--------------------------------------------------------
commercialiser3 0 commercialiser2 6 cuir3      39
cuir2          22 industrie3     17 industrie2 17
marche3        11 marche2        11 vendre2     6


Queries CombinedNoRep was best, unweighted.

Query 12
--------------------------------------------------------
Run Result
--------------------------------------------------------
Combined      2 CombinedNoRep 13
--------------------------------------------------------
agriculture3 15 agriculture2  13 commerce3      0
commerce2     3 echanger2     12 impact3        5
impact2      10 international3 5 international2 5
metier3       7 metier2       13 organique3    10
organique2   12

Query 17
--------------------------------------------------------
Run Result
--------------------------------------------------------
Combined      22 CombinedNoRep  24
--------------------------------------------------------
agriculture3  12 agirculture2   24 alimentaire3 24
alimentaire2  24 article3        8 article2     11
consommation3 21 consommation2  24 information3  4
information2  11 pomme3         65 pomme2       43
recherche4     4 recherche3      6 recherche2   11
renseignement3 4 renseignement2 11 terre3       38
terre2        35

Query 41
--------------------------------------------------------
Run Result
--------------------------------------------------------
Combined    11 CombinedNoRep               19
--------------------------------------------------------
allemagne3  28 allemagne2  28 consequence2 15
difficulte3  7 difficulte2 12 entrainer3    3
entrainer2   9 militaire3  26 militaire2   26
necessiter3 10 necessiter2 10 prestige3    13
prestige2   19 situation3   7 situation2   12
standing2   19 unifie3     19 unifie2      19

Query 50
--------------------------------------------------------
Run Result
--------------------------------------------------------
```

```
Combined   6 CombinedNoRep        10
-------------------------------------------------------
accident3 12 accident2 12 chemin3  0
chemin2    3 conduire2  9 hasard2 10
principal3 8 principal2 9 rir2    10
route3    12 route2    12 voie3    1
voie2      3


Queries where Combined was best, weighted.

Query 3
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined       10 CombinedNoRep  6
-------------------------------------------------------
arreter3       13 arreter2      12 circulation3 0
circulation2    0 contenir3      1 contenir2    1
drogue3        15 drogue2       12 endiguer2    0
international3 11 international2 9 medicament3  0
medicament2     0 mesure3        0 mesure2      0
mesurer3        0 mesurer2       0 pied3        0
pied2           0 tige3          4 tige2        4
trafic3         9 trafic2       10

Query 4
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined    11 CombinedNoRep 7
-------------------------------------------------------
ordure3     13 ordure2      11 possibilite3 4
possibilite2 7 reusage3      7 reusage2     7

Query 16
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined     44CombinedNoRep 33
-------------------------------------------------------
monde3       22 monde2       33 mondial3      33
mondial2     33 patrie3       0 patrie2       33
pays3        33 pays2        33 raison3       33
raison2      33 raisonner2   22 reapparition2 22
redemarrage2 22 tuberculose3 56 tuberculose2  44

Query 58
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined       83 CombinedNoRep 50
-------------------------------------------------------
adriatique3    83 adriatique2   83 algue3            50
algue2         67 baisser2      33 concentration3    33
concentration2 33 considerable2 33 considerablement2 33
```

```
cote3          50 cote2         50 declin3          33
declin2        33 decliner3     33 decliner2        50
grand3         33 grand2        33 gros3            33
gros2          33 italien3      50 italien2         50
lourd3         33 lourd2        33 plage3           33
plage2         33 refuser3      50 refuser2         50
rouler2        33 tourisme3     67 tourisme2        67
violent2       33
```

Queries where CombinedNoRep was best, weighted.

Query 15

```
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined  0 CombinedNoRep       3
-------------------------------------------------------
penalite2 3 penalty3 3 penalty2 3
souci3    0 souci2   2
```

Query 24

```
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined 42 CombinedNoRep             67
-------------------------------------------------------
avancer3 42 avancer2    50 cote2      50
etendre2 50 gagner3     42 gagner2    50
gain3    42 gain2       50 grand3     58
grand2   67 large3      42 large2     50
monde4   33 monde3      42 monde2     50
mondial3 33 mondial2    50 ours5      50
ours4    50 ours3       50 ours2      67
peluche2 83 popularite2 50 porter3    50
porter2  58 supporter3  33 supporter2 50
vaste3   33 vaste2      50
```

Query 35

```
-------------------------------------------------------
Run Result
-------------------------------------------------------
Combined     15CombinedNoRep 20
-------------------------------------------------------
barriere3    15 barriere2    15 commerce3      25
commerce2    15 communaute3  20 communaute2    20
ec3           5 ec2          20 echanger2      20
europeen4    20 europeen3    15 europeen2      15
importation3 20 importation2 20 importer3      10
importer2    15 metier3       5 metier2        20
restriction3 20 restriction2 20 signification2 20
```

Query 78

```
-------------------------------------------------------
Run Result
-------------------------------------------------------
```

```
Combined     0 CombinedNoRep           13
------------------------------------------------------
economique2 13 monnaie3   0 monnaie2    0
police3      0 police2   13 politique3 13
politique2  13 rentable3  0 rentable2   0
slovenie3   13 slovenie2 25
```

Appendix 6.(i)

Results and significance tests for experiments in chapter 6 regarding equivalent S-weighting.

Run	**UnWAvP**	**WAvP**	**UnWDC20**	**WDC20**	**UnWRP**	**WRP**
CombinedNoRep	8	20	10	18	10	21
DeleteAbove1	10	19	8	14	10	19
DeleteAbove2	10	19	9	16	9	20
DeleteAbove3	10	19	10	17	12	19
DeleteAbove4	10	18	10	16	11	18
DeleteAbove5	9	19	9	17	10	20
DeleteAbove7	9	19	10	17	10	20
DeleteAbove9	8	20	10	17	10	22
DeleteAbove11	8	21	10	19	10	22
DeleteAbove13	9	21	11	19	11	24
DeleteAbove15	9	21	11	19	11	24

Figure 1: Deletion of Equivalents of Degree of Ambiguity Greater than Threshold

Paired T-Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
DeleteAbove1 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
DeleteAbove2 v. *CombinedNoRep*	0.10	0.01	0.12	0.09
DeleteAbove3 v. *CombinedNoRep*	0.89	0.21	0.79	0.35
DeleteAbove4 v. *CombinedNoRep*	0.57	0.66	0.78	0.58
DeleteAbove5 v. *CombinedNoRep*	0.54	0.69	0.32	0.66
DeleteAbove7 v. *CombinedNoRep*	0.10	0.22	0.73	0.54
DeleteAbove9 v. *CombinedNoRep*	0.40	0.11	0.50	0.87
DeleteAbove11 v. *CombinedNoRep*	0.01	0.0	0.28	0.03
DeleteAbove13 v. *CombinedNoRep*	0.01	0.0	0.29	0.0
DeleteAbove15 v. *CombinedNoRep*	0.02	0.01	0.57	0.01

Figure 2: Deletion of Equivalents, Paired T-test

Sign Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
DeleteAbove1 v. *CombinedNoRep*	0.00	0.01	0.01	0.02
DeleteAbove2 v. *CombinedNoRep*	0.11	0.21	0.20	0.29
DeleteAbove3 v. *CombinedNoRep*	0.64	0.06	0.89	0.42
DeleteAbove4 v. *CombinedNoRep*	0.29	0.17	0.36	0.33
DeleteAbove5 v. *CombinedNoRep*	0.72	1.0	0.33	0.77
DeleteAbove7 v. *CombinedNoRep*	0.28	0.35	0.74	0.65
DeleteAbove9 v. *CombinedNoRep*	0.21	0.27	0.50	0.87
DeleteAbove11 v. *CombinedNoRep*	0.04	0.0	0.01	0.38
DeleteAbove13 v. *CombinedNoRep*	0.01	0.0	0.0	0.3
DeleteAbove15 v. *CombinedNoRep*	0.0	0.01	0.02	0.58

Figure 3: Deletion of Equivalents, Sign Test

T-weight = 2.0

Run	**UnWAvP**	**WAvP**	**UnWDC20**	**WDC20**	**UnWRP**	**WRP**
CombinedNoRep	8	20	10	18	10	21
Threshold1	13	25	13	20	14	25
Threshold2	12	23	13	21	12	24
Threshold3	12	22	12	20	13	24
Threshold4	11	21	11	18	12	23
Threshold5	10	21	11	18	11	22
Threshold7	9	20	11	18	11	22
Threshold9	9	20	11	19	11	22
Threshold11	9	21	10	19	11	22
Threshold13	9	21	11	19	11	23
Threshold15	9	21	10	19	11	23

Figure 4: Applying a T-weight of 2.0 to Equivalents of Degree of Ambiguity Below or Equal to Threshold

Paired T-Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Threshold1 v. *CombinedNoRep*	0.0	0.08	0.01	0.43
Threshold2 v. *CombinedNoRep*	0.0	0.18	0.39	0.88
Threshold3 v. *CombinedNoRep*	0.0	0.04	0.01	0.29
Threshold4 v. *CombinedNoRep*	0.0	0.04	0.21	0.33
Threshold5 v. *CombinedNoRep*	0.0	0.03	0.16	0.13
Threshold7 v. *CombinedNoRep*	0.0	0.17	0.59	0.01
Threshold9 v. *CombinedNoRep*	0.0	0.01	0.55	0.22
Threshold11 v. *CombinedNoRep*	0.0	0.01	0.18	0.07
Threshold13 v. *CombinedNoRep*	0.0	0.01	0.06	0.03
Threshold15 v. *CombinedNoRep*	0.01	0.02	0.48	0.17
Threshold7 v. Threshold9	0.80	0.16	0.37	0.82
Threshold9 v. Threshold11	0.62	0.16	0.18	0.32
Threshold11 v. Threshold13	0.48	0.08	0.77	0.78
Threshold13 v. Threshold15	0.03	0.32	0.05	0.10

Figure 5: Applying a T-weight of 2.0 to Less Ambiguous Equivalents - Paired T-Test

Sign Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Threshold1 v. *CombinedNoRep*	0.12	0.32	0.01	0.52
Threshold2 v. *CombinedNoRep*	0.02	0.56	0.06	1.0
Threshold3 v. *CombinedNoRep*	0.0	0.06	0.01	0.18
Threshold4 v. *CombinedNoRep*	0.0	0.07	0.07	0.3
Threshold5 v. *CombinedNoRep*	0.01	0.01	0.22	0.13
Threshold7 v. *CombinedNoRep*	0.0	0.01	0.60	0.30
Threshold9 v. *CombinedNoRep*	0.0	0.04	0.28	0.42
Threshold11 v. *CombinedNoRep*	0.0	0.0	0.27	0.08
Threshold13 v. *CombinedNoRep*	0.0	0.0	0.07	0.05
Threshold15 v. *CombinedNoRep*	0.0	0.01	0.14	0.33

Figure 6: Applying a T-weight of 2.0 to Less Ambiguous Equivalents - Sign Test

T-weight = 3.0

Run	**UnWAvP**	**WAvP**	**UnWDC20**	**WDC20**	**UnWRP**	**WRP**
CombinedNoRep	8	20	10	18	10	21
Threshold1	14	24	12	19	16	24
Threshold2	13	22	13	20	14	24
Threshold3	13	22	13	19	14	22
Threshold4	12	20	11	18	12	21
Threshold5	11	20	11	18	12	21
Threshold7	10	20	11	18	11	21
Threshold9	9	21	11	19	11	22
Threshold11	10	21	11	19	12	23
Threshold13	9	21	11	19	11	24
Threshold15	9	21	10	19	11	23

Figure 7: Applying a T-weight of 3.0 to Equivalents of Degree of Ambiguity Below or Equal to Threshold

Paired T-Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Threshold1 v. *CombinedNoRep*	0.0	0.41	0.01	0.65
Threshold2 v. *CombinedNoRep*	0.79	0.07	0.78	0.03
Threshold3 v. *CombinedNoRep*	0.0	0.36	0.03	0.58
Threshold4 v. *CombinedNoRep*	0.0	0.42	0.16	0.88
Threshold5 v. *CombinedNoRep*	0.0	0.12	0.13	0.87
Threshold7 v. *CombinedNoRep*	0.0	0.04	0.73	0.73
Threshold9 v. *CombinedNoRep*	0.0	0.04	0.61	0.58
Threshold11 v. *CombinedNoRep*	0.0	0.02	0.13	0.02
Threshold13 v. *CombinedNoRep*	0.0	0.0	0.06	0.01
Threshold15 v. *CombinedNoRep*	0.0	0.0	0.48	0.03

Figure 8: Applying a T-weight of 3.0 to Less Ambiguous Equivalents - Significance Tests

Sign Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Threshold1 v. *CombinedNoRep*	0.0	0.21	0.0	0.72
Threshold2 v. *CombinedNoRep*	0.0	0.48	0.0	0.53
Threshold3 v. *CombinedNoRep*	0.01	0.35	0.04	0.58
Threshold4 v. *CombinedNoRep*	0.01	0.49	0.21	1.0
Threshold5 v. *CombinedNoRep*	0.0	0.20	0.12	0.83
Threshold7 v. *CombinedNoRep*	0.0	0.03	0.73	1.0
Threshold9 v. *CombinedNoRep*	0.0	0.11	0.61	0.58
Threshold11 v. *CombinedNoRep*	0.0	0.0	0.19	0.02
Threshold13 v. *CombinedNoRep*	0.0	0.0	0.07	0.01
Threshold15 v. *CombinedNoRep*	0.0	0.0	0.50	0.05

Figure 9: Applying a T-weight of 3.0 to Less Ambiguous Equivalents - Sign Test

T-weight = 4.0

Run	**UnWAvP**	**WAvP**	**UnWDC20**	**WDC20**	**UnWRP**	**WRP**
CombinedNoRep	8	20	10	18	10	21
Threshold1	15	23	14	18	17	23
Threshold2	13	22	13	19	14	23
Threshold3	13	21	13	19	14	21
Threshold4	13	20	11	17	13	20
Threshold5	11	20	11	18	12	20
Threshold7	10	19	10	18	10	20
Threshold9	9	20	11	18	11	22
Threshold11	9	21	11	19	12	22
Threshold13	9	21	11	19	11	24
Threshold15	9	21	10	19	11	23

Figure 10: Applying a T-weight of 4.0 to Equivalents of Degree of Ambiguity Below or Equal to Threshold

Paired T-Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Threshold1 v. *CombinedNoRep*	0.0	1.0	0.01	0.77
Threshold2 v. *CombinedNoRep*	0.0	0.7	0.05	0.67
Threshold3 v. *CombinedNoRep*	0.0	0.60	0.02	0.48
Threshold4 v. *CombinedNoRep*	0.0	0.80	0.11	0.89
Threshold5 v. *CombinedNoRep*	0.0	0.18	0.13	0.54
Threshold7 v. *CombinedNoRep*	0.0	0.09	0.73	0.87
Threshold9 v. *CombinedNoRep*	0.0	0.04	0.61	0.59
Threshold11 v. *CombinedNoRep*	0.0	0.0	0.13	0.06
Threshold13 v. *CombinedNoRep*	0.0	0.0	0.13	0.01
Threshold15 v. *CombinedNoRep*	0.0	0.0	0.48	0.02
W2.0N13 v. W3.0N13	0.32	0.18	1.0	0.06
W2.0N13 v. W4.0N13	0.16	0.26	0.32	0.02
W3.0N13 v. W4.0N13	1.0	0.32	0.32	0.16
Delete13 v. W2.0N13	0.53	0.25	0.71	0.02
Delete13 v. W3.0N13	0.74	1.0	0.71	0.10
Delete13 v. W4.0N13	0.74	0.66	0.42	0.16
W2.0N15 v. W3.0N15	0.32	0.05	1.0	0.02
W2.0N15 v. W4.0N15	0.32	0.03	1.0	0.02
W3.0N15 v. W4.0N15	1.0	1.0	1.0	1.0
Delete15 v. W2.0N15	1.0	0.13	1.0	0.01
Delete15 v. W3.0N15	1.0	0.58	1.0	0.26
Delete15 v. W4.0N15	1.0	1.0	1.0	0.18

Figure 11: Applying a T-weight of 4.0 to Less Ambiguous Equivalents - Paired T-Test

Sign Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Threshold1 v. *CombinedNoRep*	0.16	1.0	0.02	0.88
Threshold2 v. *CombinedNoRep*	0.07	0.64	0.07	0.78
Threshold3 v. *CombinedNoRep*	0.01	1.0	0.03	0.67
Threshold4 v. *CombinedNoRep*	0.0	0.82	0.15	1.0
Threshold5 v. *CombinedNoRep*	0.0	0.24	0.18	0.64
Threshold7 v. *CombinedNoRep*	0.0	0.15	0.74	1.0
Threshold9 v. *CombinedNoRep*	0.0	0.21	0.61	0.86
Threshold11 v. *CombinedNoRep*	0.0	0.19	0.0	0.06
Threshold13 v. *CombinedNoRep*	0.0	0.0	0.15	0.01
Threshold15 v. *CombinedNoRep*	0.0	0.01	0.50	0.03

Figure 12: Applying a T-weight of 4.0 to Less Ambiguous Equivalents - Sign Test

Appendix 6.(ii)

Results and significance tests regarding experiments in chapter 6 on query term Q-weighting.

Paired T-Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
DeleteAbove1 v. *CombinedNoRep*	0.31	0.69	0.29	0.50
DeleteAbove2 v. *CombinedNoRep*	0.17	0.55	0.80	0.63
DeleteAbove3 v. *CombinedNoRep*	0.62	0.23	0.80	0.28
DeleteAbove4 v. *CombinedNoRep*	0.04	0.56	0.22	0.24
DeleteAbove5 v. *CombinedNoRep*	0.52	0.53	0.56	0.90
DeleteAbove7 v. *CombinedNoRep*	0.02	0.10	0.05	0.05
DeleteAbove9 v. *CombinedNoRep*	0.0	0.0	0.01	0.03
DeleteAbove11 v. *CombinedNoRep*	0.0	0.0	0.06	0.23
DeleteAbove13 v. *CombinedNoRep*	0.0	0.09	0.52	0.55
DeleteAbove15 v. *CombinedNoRep*	0.03	0.11	0.80	0.44

Figure 13: Deletion of Highly Ambiguous Terms - Paired T-Test

Sign Test	UnW AvP	W AvP	UnW RP	W RP
DeleteAbove1 v. *CombinedNoRep*	0.43	0.60	0.23	0.68
DeleteAbove2 v. *CombinedNoRep*	0.0	0.0	0.08	0.29
DeleteAbove3 v. *CombinedNoRep*	0.47	0.64	1.0	0.63
DeleteAbove4 v. *CombinedNoRep*	0.12	0.50	0.28	0.42
DeleteAbove5 v. *CombinedNoRep*	1.0	0.56	0.66	1.0
DeleteAbove7 v. *CombinedNoRep*	0.21	0.04	0.07	0.07
DeleteAbove9 v. *CombinedNoRep*	0.01	0.0	0.01	0.05
DeleteAbove11 v. *CombinedNoRep*	0.0	0.0	0.08	0.29
DeleteAbove13 v. *CombinedNoRep*	0.02	0.03	0.66	0.69
DeleteAbove15 v. *CombinedNoRep*	0.24	0.07	1.0	0.45

Figure 14: Deletion of Highly Ambiguous Terms - Sign Tests

Q-weight = 2.0

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
Threshold1	12	24	13	21	15	26
Threshold2	14	27	14	22	16	29
Threshold3	15	27	14	21	17	28
Threshold4	15	26	14	21	17	28
Threshold5	14	25	12	21	15	27
Threshold7	13	25	13	21	15	27
Threshold9	11	25	13	21	12	27
Threshold11	10	25	12	21	12	26
Threshold13	10	24	12	20	11	25
Threshold15	9	22	11	19	10	22

Figure 15: Applying a Q-weight of 2.0 to Terms of Degree of Ambiguity Below or Equal to Threshold

Paired T-Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Thresh1W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh2W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh3W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh4W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh5W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.02
Thresh7W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh9W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh11W2.0 v. *CombinedNoRep*	0.0	0.0	0.01	0.0
Thresh13W2.0 v. *CombinedNoRep*	0.0	0.0	0.16	0.02
Thresh15W2.0 v. *CombinedNoRep*	0.0	0.0	0.80	0.02
Thresh1W2.0 v. Thresh2W2.0	0.13	0.16	0.47	0.04
Thresh2W2.0 v. Thresh3W2.0	1.0	0.55	0.84	1.0
Thresh3W2.0 v. Thresh4W2.0	0.32	0.52	0.32	0.05
Thresh4W2.0 v. Thresh5W2.0	0.03	0.03	0.06	0.02
Thresh5W2.0 v. Thresh7W2.0	0.11	0.37	0.82	0.25
Thresh7W2.0 v. Thresh9W2.0	0.04	0.24	0.0	0.29
Thresh9W2.0 v. Thresh11W2.0	0.13	0.21	0.10	0.03
Thresh11W2.0 v. Thresh13W2.0	0.03	0.01	0.13	0.04
Thresh13W2.0 v. Thresh15W2.0	0.05	0.13	0.05	0.13
Thresh1W2.0 v. Thresh3W2.0	0.20	0.43	0.39	0.25
Thresh3W2.0 v. Thresh5W2.0	0.0	0.02	0.01	0.0
Thresh3W2.0 v. Thresh7W2.0	0.01	0.03	0.21	0.15
Thresh3W2.0 v. Thresh9W2.0	0.01	0.01	0.0	0.08
Thresh3W2.0 v. Thresh11W2.0	0.0	0.01	0.0	0.02
Thresh3W2.0 v. Thresh15W2.0	0.0	0.0	0.80	0.02

Figure 16: Applying Q-weighting of 2.0 to Less Ambiguous Terms - Paired T-Test

Sign Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Thresh1W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh2W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh3W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh4W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh5W2.0 v. *CombinedNoRep*	0.0	0.22	0.01	0.13
Thresh7W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.01
Thresh9W2.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh11W2.0 v. *CombinedNoRep*	0.0	0.02	0.0	0.0
Thresh13W2.0 v. *CombinedNoRep*	0.0	0.17	0.0	0.03
Thresh15W2.0 v. *CombinedNoRep*	0.01	0.8	0.02	0.03

Figure 17: Applying Q-weighting of 2.0 to Less Ambiguous Terms - Sign Test

Q-weight = 3.0

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
Threshold1	14	24	14	19	17	25
Threshold2	17	28	16	21	19	29
Threshold3	17	28	15	21	19	29
Threshold4	17	27	15	21	18	28
Threshold5	15	26	13	21	16	27
Threshold7	14	26	14	21	15	28
Threshold9	12	25	13	21	13	28
Threshold11	11	25	13	21	13	27
Threshold13	11	24	12	20	12	26
Threshold15	9	22	11	19	11	22

Figure 18: Applying a Q-weight of 3.0 to Terms of Degree of Ambiguity Below or Equal to Threshold

Paired T-Test	UnW AvP	W AvP	UnW RP	W RP
W3.0 v. *CombinedNoRep*				
Thresh1W3.0 v. *CombinedNoRep*	0.0	0.01	0.0	0.01
Thresh2W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh3W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh4W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh5W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.01
Thresh7W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh9W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh11W3.0 v. *CombinedNoRep*	0.0	0.02	0.0	0.01
Thresh13W3.0 v. *CombinedNoRep*	0.0	0.0	0.10	0.07
Thresh15W3.0 v. *CombinedNoRep*	0.0	0.0	0.44	0.20
W2.0 v. W3.0				
Thresh1W2.0 v. Thresh11W3.0	0.0	0.73	0.06	0.66
Thresh2W2.0 v. Thresh2W3.0	0.0	0.17	0.01	0.58
Thresh3W2.0 v. Thresh3W3.0	0.0	0.41	0.04	0.52
Thresh4W2.0 v. Thresh4W3.0	0.0	0.59	0.16	0.67
Thresh5W2.0 v. Thresh5W3.0	0.0	1.0	1.0	1.0
Thresh7W2.0 v. Thresh7W3.0	0.0	0.32	1.0	0.50
Thresh9W2.0 v. Thresh9W3.0	0.0	0.14	0.05	0.20
Thresh11W2.0 v. Thresh1W3.0	0.01	0.05	0.23	0.29

Figure 19: Applying Q-weighting of 3.0 to Less Ambiguous Terms - Paired T-Test

Sign Test	UnW AvP	W AvP	UnW RP	W RP
W3.0 v. *CombinedNoRep*				
Thresh1W3.0 v. *CombinedNoRep*	0.0	0.03	0.0	0.02
Thresh2W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh3W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh4W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh5W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.01
Thresh7W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh9W3.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh11W3.0 v. *CombinedNoRep*	0.0	0.0	0.03	0.0
Thresh13W3.0 v. *CombinedNoRep*	0.0	0.01	0.15	0.12
Thresh15W3.0 v. *CombinedNoRep*	0.3	0.03	0.45	0.3

Figure 20: Applying Q-weighting of 3.0 to Less Ambiguous Terms - Sign Test

Q-weight = 4.0

Run	UnWAvP	WAvP	UnWDC20	WDC20	UnWRP	WRP
CombinedNoRep	8	20	10	18	10	21
Threshold1	15	24	14	18	18	24
Threshold2	18	27	16	21	19	28
Threshold3	18	27	16	21	20	28
Threshold4	18	27	15	21	18	28
Threshold5	16	26	13	20	16	27
Threshold7	15	26	14	20	15	27
Threshold9	12	25	13	22	13	27
Threshold11	11	25	13	21	13	26
Threshold13	11	24	12	20	12	25
Threshold15	9	22	11	19	11	23

Figure 21: Applying a Q-weight of 4.0 to Terms of Degree of Ambiguity Below or Equal to Threshold

Paired T-Test	UnW AvP	W AvP	UnW RP	W RP
W4.0 v. *CombinedNoRep*				
Thresh1W4.0 v. *CombinedNoRep*	0.0	0.03	0.0	0.03
Thresh2W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh3W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh4W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh5W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.01
Thresh7W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.02
Thresh9W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.02
Thresh11W4.0 v. *CombinedNoRep*	0.0	0.0	0.03	0.08
Thresh13W4.0 v. *CombinedNoRep*	0.0	0.01	0.10	0.40
Thresh15W4.0 v. *CombinedNoRep*	0.0	0.0	0.32	0.17
W2.0 v. W4.0				
Thresh1W2.0 v. Thresh1W4.0	0.01	0.62	0.01	0.18
Thresh2W2.0 v. Thresh2W4.0	0.0	0.56	0.0	0.74
Thresh3W2.0 v. Thresh3W4.0	0.0	0.64	0.01	0.72
Thresh4W2.0 v. Thresh4W4.0	0.0	0.63	0.15	0.71
Thresh5W2.0 v. Thresh5W4.0	0.0	0.87	0.81	1.0
Thresh7W2.0 v. Thresh7W4.0	0.0	0.39	0.81	0.67
Thresh9W2.0 v. Thresh9W4.0	0.0	0.16	0.16	1.0

Figure 22: Applying Q-weighting of 4.0 to Less Ambiguous Terms - Paired T-Test

Sign Test	UnW AvP	W AvP	UnW RP	W RP
W4.0 v. *CombinedNoRep*				
Thresh1W4.0 v. *CombinedNoRep*	0.0	0.02	0.0	0.05
Thresh2W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thres32W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh4W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh5W4.0 v. *CombinedNoRep*	0.0	0.02	0.0	0.02
Thresh7W4.0 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Thresh9W4.0 v. *CombinedNoRep*	0.0	0.0	0.01	0.02
Thresh11W4.0 v. *CombinedNoRep*	0.0	0.0	0.04	0.11
Thresh13W4.0 v. *CombinedNoRep*	0.0	0.01	0.15	0.52
Thresh15W4.0 v. *CombinedNoRep*	0.07	0.04	0.45	0.18

Figure 23: Applying Q-weighting of 4.0 to Less Ambiguous Terms - Sign Test

Appendix 6.(iii)

Results and significance tests of experiments in chapter 6 regarding stepped Q-weighting of query terms.

Significance Test	**UnW AvP**	**W AvP**	**UnW RP**	**W RP**
Stepped2 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped3 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped4 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped5 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped7 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped9 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped11 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped13 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped15 v. *CombinedNoRep*	0.0	0.0	0.0	0.0
Stepped2 v. Stepped3	0.06	0.22	0.37	0.16
Stepped3 v. Stepped4	0.71	0.75	0.81	0.45
Stepped4 v. Stepped5	1.0	1.0	1.0	1.0
Stepped5 v. Stepped7	0.06	0.0	0.05	0.02
Stepped7 v. Stepped9	0.0	0.0	0.26	0.0
Stepped9 v. Stepped11	0.0	0.03	0.26	0.37
Stepped11 v. Stepped13	0.0	0.02	0.08	0.16
Stepped13 v. Stepped15	0.10	0.03	0.16	0.03
Stepped2 v. Stepped4	0.01	0.08	0.25	0.50
Stepped 4 v. Stepped7	0.02	0.01	0.02	0.15
Stepped2 v. Thresh3W2.0	0.38	0.11	0.31	0.28
Stepped3 v. Thresh3W2.0	0.01	0.02	0.10	0.09
Stepped4 v. Thresh3W2.0	0.04	0.03	0.24	0.04
Stepped5 v. Thresh3W2.0	0.49	0.04	0.76	0.03

Figure 24: Applying Q-Weights According to Step Function - Paired T-Test