EIJA AIRIO

Morphological Problems in IR and CLIR

Applying linguistic methods and approximate string matching tools

■

ACADEMIC DISSERTATION
To be presented, with the permission of
the Faculty of Information Sciences of the University of Tampere,
for public discussion in the Auditorium Pinni B 1097,
Kanslerinrinne 1, Tampere, on June 13th, 2009, at 12 o'clock.

UNIVERSITY OF TAMPERE

ACADEMIC DISSERTATION
University of Tampere
Department of Information Studies and Interactive Media
Finland

Distribution
Bookshop TAJU
P.O. Box 617
33014 University of Tampere
Finland

Tel. +358 3 3551 6055
Fax +358 3 3551 7685
taju@uta.fi
www.uta.fi/taju
http://granum.uta.fi

Cover design by
Juha Siro

Acta Universitatis Tamperensis 1414
ISBN 978-951-44-7707-2 (print)
ISSN-L 1455-1616
ISSN 1455-1616

Acta Electronica Universitatis Tamperensis 842
ISBN 978-951-44-7708-9 (pdf)
ISSN 1456-954X
http://acta.uta.fi

Tampereen Yliopistopaino Oy – Juvenes Print
Tampere 2009

Kiitokset

Väitöskirjan kirjoittaminen työn ohessa on mielestäni ollut hieno kokemus ja mahtava mahdollisuus. En epäröinyt, kun minulle tarjottiin tätä mahdollisuutta aloittaessani laboratorioinsinöörin viransijaisena informaatiotutkimuksen laitoksella vuonna 2001. Olen iloinen, että otin haasteen vastaan. Oman tutkimuksen tekeminen on ollut palkitseva ja mielenkiintoinen prosessi.

Minulla on ollut kaksi ohjaajaa tukemassa työtäni: Kalervo Järvelin ja Jaana Kekäläinen, joille haluan osoittaa suuret kiitokset. Kaikkein mukavimmat muistot väitöskirjaprosessissa liittyvät ohjaajieni kanssa käytyihin keskusteluihin. Oli hienoa, kun kaksi professoria ja alan asiantuntijaa perehtyi teksteihini, kommentoi niitä ja teki parannusehdotuksia. Ja kaiken tämän lisäksi nuo ohjaustilaisuudet olivat hauskoja!

Informaatiotutkimuksen laitoksella (nykyisin informaatiotutkimuksen ja interaktiivisen median laitos) toimii tiedonhaun tutkimusryhmä FIRE (Finnish Information Retrieval Expert Group). FIRE:n seminaareissa olen sekä saanut rakentavaa palautetta tutkimuksestani että päässyt lukemaan ja kommentoimaan kollegojen tutkimusta. Myös lounasseura on ollut tärkeässä roolissa väitöskirjaprosessissani antamalla sekä muuta ajateltavaa että uskoa omiin kykyihin. Haluan kiittää tuesta ja toveruudesta kaikkia edellä mainittuihin ryhmiin kuuluvia henkilöitä, samoin kuin muitakin ihmisiä informaatiotutkimuksen laitoksella.

Lopuksi osoitan kiitokseni perheelleni, joka on aina kannustanut minua kaikissa pyrkimyksissäni: miehelleni Erkille ja lapsillemme Ellalle, Eskolle ja Eemelille.

Kalliossa 2.5.2009

Eija Airio

Abstract

The topics of the present thesis are linguistic and approximate string matching methods in monolingual, bilingual and multilingual information retrieval. The linguistic approaches applied in the studies are word normalization, translation and word form generation, while n-gramming and s-gramming represent approximate string matching techniques.

The first contribution of this thesis is connected to compounds: we studied the importance of index decompounding in mono- and bilingual retrieval. The impact of decompounding, especially on bilingual retrieval, is not a very widely studied issue in IR literature. Index decompounding did not have any notable impact on monolingual retrieval. On the other hand, we found that in bilingual retrieval, index decompounding is vital, when the source language is phrase oriented while compounds are used in the target language.

The second contribution deals with the quality of translation dictionaries. We found that the quality of a dictionary has an even larger effect on the bilingual retrieval result than has been supposed. We also performed user tests in bilingual retrieval utilizing dictionaries of various quality. We found that query translation is generally beneficial for users with moderate or poor language skills, but only if the translation dictionary is of good quality: a defective dictionary does not help even those with poor target language skills.

The third contribution of this thesis is connected with bilingual user tests. In prior research, the performance of bilingual retrieval compared with monolingual retrieval has mostly been tested only in a laboratory environment. We found that query translation performed much better in user tests than it has performed in laboratory tests. The reason is that the target queries formulated by test persons (and in a real CLIR situation) are often defective, because they are formulated by people with only moderate or poor language skills, while laboratory tests utilize queries formulated by native language speakers.

Our fourth contribution is connected to bilingual retrieval in an inflected index: many IR indexes are non-normalized, and thus, there is a need for a practical method to perform bilingual retrieval in an inflected word form index. There is not much research on the issue, however. We found that any kind of processing (approximate string matching, frequent case generation or their combination) improves the retrieval result compared with queries formulated directly from raw translations. This information may be important when developing systems utilizing bilingual retrieval.

Fifth, we found that various normalization approaches in indexing and retrieval do not have any remarkable impact on the multilingual retrieval results,

even if lemmatization seems to perform slightly better than stemming. Various result list merging approaches have only a minor impact on the result. On the other hand, there are not many systems utilizing separate indexes with result list merging. Thus, multilingual IR research should be directed towards systems with a merged index.

List of figures

Original research publications

This thesis consists of a summary and the following original studies.

I. Eija Airio, Heikki Keskustalo, Turid Hedlund & Ari Pirkola. 2003. UTACLIR @ CLEF 2002 – Bilingual and multilingual runs with a unified process. In Peters, Kluck, Gonzalo: Advances in cross-language information retrieval. Results of the Cross-Language Evaluation Forum – CLEF 2002. Lecture Notes in Computer Science 2785, Springer Verlag , 91-100. Reproduced here by permission.

II. Eija Airio, Heikki Keskustalo, Turid Hedlund & Ari Pirkola. 2004. The impact of word normalization methods and merging strategies on multilingual IR. In Peters, Gonzalo, Braschler, Kluck: Comparative Evaluation of multilingual information access systems. Lecture notes in Computer Science 3237, Springer Verlag, 74-84. Reproduced here by permission.

III. Eija Airio. 2006. Word normalization and decompounding in mono- and bilingual IR. Information Retrieval 9(3), 249-271. Reproduced here by permission.

IV. Eija Airio & Kimmo Kettunen. 2008. Does dictionary based bilingual retrieval work in a non-normalized index? Accepted for publication in Information Processing & Management with minor revisions.

V. Eija Airio. 2008. Who benefits from CLIR in Web retrieval? Journal of Documentation 64(5), 760-778. Reproduced here by permission.

The studies will be referred to as Study I-V in the introductory part of the thesis.

1 Introduction

Information retrieval (IR) is concerned with information needs of users of IR systems. From an IR point of view, information is located in documents (text, image, sound or some other type of documents): for example newspaper articles, photos or pieces of music. The task of IR is to represent, store and organize documents for supporting access to them.

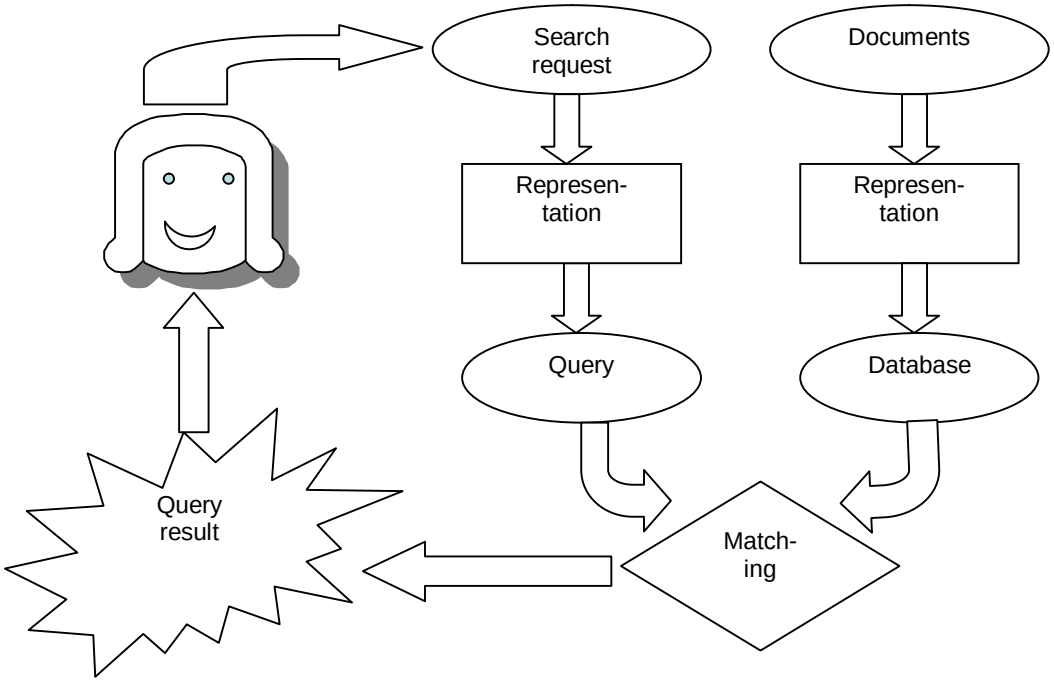

Figure 1. *An IR system (adapted from Ingwersen & Järvelin 2005, 5)*

Thus, there is a document collection and a user who wants to access documents (or information present in them). What is needed is an IR system: it facilitates storing and retrieval of documents (or their descriptions). Retrieval is based on an index. Thus, the document collection must be indexed to make it retrievable. The classic indexing approach is manual: each document is described by a set of representative keywords called **index terms**. Automatic indexing is more common today: the indexing system automatically selects the **index words**

based on the documents. [1] The user who wants to access documents has an information need, which he presents as an **information request** or a **user query** to the system. The words expressed in the query are called **query words** hereinafter. The retrieval system performs matching: it checks which documents match the query according to given rules. Finally, the system presents to the user a list of retrieved documents (see Figure 1).

A crucial part of a retrieval system is a search engine, which performs matching between a user query and documents. Search engines are based on various principles called retrieval models. There are three classical information retrieval models: the Boolean, the vector space and the probabilistic model. In the **Boolean model**, queries are presented as Boolean logic expressions. Documents either logically match the query or do not match the query at all, and only matching documents are presented to the user. The documents are presented in an arbitrary order, or for example ordered alphabetically by their header text, or chronologically. The aim of the vector space model and the probabilistic model is not only to judge documents as wholly matching or not matching the query. Instead, documents are given scores depending on their degree of matching. Thus, some documents match the query better than the others. Retrieved documents are represented to the user as a ranked list in a descending order according to the score. Systems based on the **vector space model** represent documents and queries as vectors. The degree of similarity is concluded by calculating the angle between the query vector and each document vector. The **probabilistic model** tries to estimate the probability that a document is relevant for the user request. It is based on the assumption that the probability depends on the query and the document representations only. (Baeza-Yates & Ribeiro-Neto 1999, 24-31.)

The manual indexing approach, which was popular especially in the sixties, has some pros and cons compared to the automatic indexing approach. The index size is much smaller when utilizing manual indexing than with automatic indexing. This was an important issue some decades ago, when storage devices were more expensive than today and their storage capacity was small. On the other hand, manual indexing is dependent on human interpretation of documents. The interpretation of a person who is indexing a document may be quite different from that of a user. Also, manual indexing is based on a predefined set of index terms (for example in a thesaurus), which may be inadequate.

Automatic indexing does not suffer from the problems caused by predefined index terms. Instead, natural language poses challenges for automatic indexing. One of the most evident problems is word inflection. For example the English word *house* has two inflected forms *house* and *houses.* Thus, when applying automatic indexing, there might be two index entries for the word *house*. To retrieve all the documents containing the word *house*, both *house* and *houses* should be included in the query. More generally, to retrieve documents

[1] *Here and hereinafter, when discussing documents and document collections, we are referring to text documents.*

containing a word, all inflected forms representing the word should be included in the query. In some languages, for example English, inflection is quite weak: nouns have only two cases, and they are inflected in singular and plural. There are several languages, which have much stronger inflection than English has. In Finnish, for example, nouns have 14 morphological cases, and they are inflected in singular and plural in all the cases, and personal and other suffixes may be added to all these.

Natural language processing (NLP) approaches and tools can be utilized in IR to overcome word inflection and other problems alike. The IR approach utilizing linguistic tools for document and query pre-processing is called **linguistic IR.** An NLP approach for diminishing the effects of word inflection is called word **normalization**. The normalization approaches may be divided into two groups: lemmatization and stemming. In an ideal normalization case, all the inflected forms of the same word (representing an index word) would have a common index entry. This kind of index is called a normalized index. In our example, words *house* and *houses* would have one entry (for example *house* or *hous*). The query should naturally contain normalized word forms as well. The normalization approach can be also called the **reductive** approach, because the aim is to reduce all the inflected word forms into one form. The opposite approach is called the **generative** approach: given the lemma of a word, inflected word forms are generated. (See Kettunen 2006.)

Word normalization is not always possible: the tools needed might not be available or a tool might not be able to normalize a given string. **Approximate string matching techniques** are often utilized in IR and CLIR in addition to word normalization tools. The aim of approximate string matching is to find the best matching strings for a given string: for example to find the best index matches for a query word which a normalization tool is not able to handle.

In monolingual IR, queries and documents are represented in the same language. It might happen, however, that a user is not able to express his query in the document language, even if he is able to read documents. It is also possible, that a user is performing her retrieval in a multilingual collection (for example in the Web), and would like to retrieve documents in multiple languages by expressing a query in a single language. **Cross language information retrieval (CLIR)** is useful for both of these user scenarios. In CLIR, the query language is called the **source language**, and the document language(s) the **target language(s)**. The CLIR tasks are either **bilingual** or **multilingual**. In a bilingual task, the document collection includes documents in one language only (thus there is only one target language). Bilingual IR is useful for the former user in the examples above. **Multilingual IR (MLIR)** is concerned with multilingual collections (and several target languages). Thus, it benefits the latter user.

CLIR is based on translation. Translation systems and word normalization systems are often based on dictionaries. Dictionaries always include a limited number of words, while natural languages evolve all the time, and new words are generated. Thus, no dictionary can include all the possible words of a language.

The words not included in a dictionary are called **OOV (**out-of-vocabulary) **words**.

The multilingual task based on query translation is more complicated than the bilingual one, because there are several target languages. If the indexes for the target languages are separate, the only approach is to execute several bilingual tasks, and finally **merge the results**. If we have an Internet type multilingual target index, we have two possibilities: either to perform separate retrievals and merge the results, or to perform a merged, multilingual retrieval.

IR offers several research scenarios connected for example with retrieval, interfaces or user behaviour. Even if the core of IR is to find information, some of the research interests are not directly connected with human beings. Thus, IR research has two main foci: the laboratory-based focus and the user-based focus. The main interests of the former are building up efficient indexes, processing queries, and developing ranking algorithms for the result set. The human-centred IR concentrates on studying the behaviour of the user, and understanding his needs. (Baeza-Yates & Ribeiro-Neto 1999, 7.)

Traditional IR research has a laboratory-based focus, and it is based on laboratory tests: there is a test database, test topics and relevance assessments for the topics. This is a reliable and static model, facilitating comparisons of separate tests of various research groups at various points in time. The validity of laboratory testing is not evident, however: laboratory tests measure system performance without human intervention. Thus, also **user tests** are necessary in order to test the usability of an approach or a system for potential users.

The present thesis investigates linguistic and approximate string matching methods for controlling morphological variation in IR and CLIR. Linguistic methods cover for example word normalization and translation. String matching methods are based on similarity of strings: they are thus language-independent methods. We shall look at the following research questions:

Morphological and approximate string matching methods in IR and CLIR:

1. What is the impact of approximate string matching of OOV words on monolingual (Finnish) retrieval?
2. What is the impact of the operator utilized to envelope parts of compounds on monolingual (Finnish) retrieval?
3. What is the impact of alternative translation dictionaries on bilingual (English-Finnish) retrieval?
4. Which normalization and result list merging approach combination performs best in multilingual (English -> Dutch, English, Finnish, French, German, Italian, Spanish, Swedish) retrieval?
5. Which word normalization method gives the best result in monolingual (English, Finnish, German and Swedish) retrieval?
6. Which word normalization method gives the best result in bilingual (English-Finnish, English-German and English-Swedish) retrieval?

Morphological and approximate string matching methods in bilingual retrieval in an inflected index:

7. Which of the following inflected methods performs best in bilingual retrieval (English-Finnish, Swedish-Finnish, English-Swedish, Finnish-Swedish) in a non-normalized index: approximate string matching methods or generative methods, or a combination of those?
8. What are the reasons for performance differences between various methods in distinct language pairs?
9. Is the result based on the best inflected method commensurate with that of the gold standard?

Morphological methods in interactive CLIR:

10. Which performs better, dictionary-based translations (Finnish-Swedish, English-German, Finnish-French) or the user formulated target language queries (Swedish, German, French)?
11. To what degree do the following user characteristics affect the IR performance achieved through the translated queries / the target language queries: the language skills, the topic domain familiarity, the topic vocabulary familiarity?

The research questions 1, 2 and 3 are discussed in Study I, while Study II answers the fourth research question. Study III deals with the research questions 5 and 6, while Study IV answers the research questions 7, 8 and 9. Study V deals with the research questions 10 and 11.

The rest of this thesis is organized in the following way. Natural language and its influence on IR are discussed in Chapter two. Chapter three introduces cross-language IR. Chapter four discusses approaches to IR and CLIR research and evaluation. Chapter five includes the summary of the studies, and Chapter six closes the thesis with discussion and conclusions, followed by the contributing articles.

2 Natural language in IR and CLIR

2.1 Natural language concepts and features

Natural languages are constantly evolving, quite vague means of communication. Rather than being a demerit, vagueness tells about adaptability and power of expression of natural languages (Karlsson 1994, 3). Because of adaptability of natural languages, introducing new concepts, as well as using old concepts in new contexts is possible. Naturally, due to vagueness, meanings of natural language expressions are not stable. For example, the meaning of words may broaden, narrow or change according to human needs (Akmajian & al. 1995, 85-87).

Natural languages can be seen as structured systems. There are five sub-systems which construct a language: semantic, phonologic, lexical, morphological and syntactic sub-systems. **Semantics** deals with the meaning, while **phonology** deals with phonetic forms of languages. The **lexical** system is constructed of the words of a language. **Morphology** deals with the internal structure of words, and **syntax** with the structure and the rules of sentences. Of these sub-systems, phonologic, lexical, morphological and syntactic subsystems are **tangible** or **linguistic** systems, while **semantics** is **intangible** dealing with human meaning construction processes. (Karlsson 1994, 14-15.)

All natural languages are constructed of words and sentences. A word can be 1) a lexical word, that is the word in its basic form (a cat), 2) a word in its inflected form (cats), 3) a derived word (catty), or 4) a word consisting of two or more single words (a catfish). Words 1-3 are all **irreducible** words: they contain only one "autonomous" word (Karlsson 1994, 75-76). Number 4) is sometimes called **a compound**, but there are also broader definitions for a compound. Compounds can be divided in three types: 1) the **closed form**, where the parts of a compound are written together (makeup), 2) the **hyphenated form**, where the parts are attached by a hyphen (daughter-in-law), and 3) the **open form** (post office). In the present research, the closed form and the hyphenated form are called compounds, while the open form is called **a phrase**. Compounds of this definition are quite rare in English, while for example Finnish, Swedish and German have a lot of them. Phrases are used in English instead of compounds.

The present thesis concentrates on morphological aspects of natural languages. The most crucial morphological concepts for this research are a morpheme, a stem and a root (also called a linguistic root). A **morpheme** is the smallest linguistic unit which has an independent meaning or at least a linguistic

function. For example the word *unbelievable* has three morphemes: *un*, *believ* and *able*. A **stem** is the base part of a word not including inflectional morphemes. The stem for the word *cats* is *cat*, and the stem of the word *unbelievable* is *unbeliev*. A **root** is the minimal unit of the stem representing the semantic content of the word – thus, it cannot be split into smaller parts. (Karlsson 1994, 100-101.) In English, the root and the stem are often the same. The root of the word *cats* is *cat*, and the root of the word *unbelievable* is *believ*.

There are two schools of thought in the IR field with respect to the choice of the language level and language processing in indexing and retrieval. One prefers the semantic level approach. This approach could be called **conceptual information retrieval**. Semantic processing requires large amounts of pre-coded knowledge. The other school of thought bases its approach on the morphologic level. It states that it is not important to understand the concepts, but locate the relevant documents. This approach is sometimes called **natural language information retrieval**, while it operates only at the morphological level. (Sheridan & Smeaton 1992.)

Morphological properties of natural languages are very diverse. Pirkola has presented a morphological classification of languages for IR purposes. It is based on two variables: the **index of synthesis** and the **index of fusion**. The index of synthesis tells the number of affixes in a language, while the index of fusion describes the ease with which affixes can be segmented in words in a language. These indicators can be used for developing and evaluation of IR systems. (Pirkola 2001.)

2.1.1 Problems caused by natural language features for IR and CLIR

The vagueness of natural languages is a problem from the IR point of view. Natural languages are flexible and constantly evolving systems: for example, a word may have various senses depending on the context and new words are created constantly. (Ingwersen & Järvelin 2005, 151.)

The meaning of a single natural language word is not necessarily precise. A word may even have several quite varied meanings. On the other hand, there might be several separate words to express the same concept. The well-known phenomenon where two or more words have an equal sense or denotation is called **synonymy** (Karlsson 1994, 217-218). Exact synonymy is quite rare, however, because words are often synonymous only in some context(s). Sometimes the word quasisynonymy is used instead of synonymy to refer to words which have the same meaning in some context. (Pirkola 1999, 23.) Synonymy causes difficulties for IR because various (synonymous) words may be used in documents to refer to the same concept. The user should supply all those words in the query in order to retrieve all the documents discussing the subject. In CLIR, the situation is more complicated. If the source language query includes all the synonyms and quasisynonyms for a word, the translated query may become too broad. On the other hand, sometimes translation acts like query

expansion in a good sense, because translation dictionaries often include synonyms for a given word.

Homonymy and polysemy are the opposites of synonymy. Homonymy and polysemy are related concepts and it is hard to make a clear distinction between them. The traditional distinction is that in **polysemy**, one word has several senses, and in **homonymy**, two different words happen to have the same form. An etymological criterion for making the distinction may be utilized: if a word has differing sound or spelling variants for its senses in the history, it is not a polysemic, but a homonymic word. The etymology of a word may be unknown, however. (Kilgarriff 1992.) In addition, this might not hold in Finnish, for example, where sound and spelling variants are rare. In Finnish, the following criterion may be used: if the inflection rules for various senses of the word are equal, the word is polysemic (Karlsson 1994, 213-214). For example a Finnish word *kieli* is a polysemic word. It means *a tongue, a language, speech, a flap, a clapper* and a *string of an instrument*. The Finnish word *kuusi* is a homonymous word. It has two meanings: *six* and *a spruce*, and their inflection rules differ. In addition, a special case for homonymy may be defined: **inflectional homonymy**. It means that some inflected forms of words happen to be identical while the lemmas are different. For example the Finnish inflected word *hauista* has three meanings: *from retrievals*, *from pikes*, *biceps* (the partitive form).

Homonymy and polysemy may cause problems for IR and CLIR: if the query includes a word with many senses, documents with any of those senses will be retrieved, even if the user would be interested in one sense only. For example if the user inputs a one word query *kuusi*, searching for information on *spruces*, he would retrieve documents including number six as well. The problem is often solved by supplying more query words, which together disambiguate the query in a natural way. (Pirkola 1999, 14.)

Problems caused by morphological features of languages are in the focus of this thesis. **Compounds** and **phrases** are problematic for IR and CLIR. The headword of a compound may be inaccessible in retrieval, which might cause loss of relevant documents. In addition, the meaning of a compound or a phrase is often more than the meaning of their constituents alone. **Word inflection** causes difficulties for IR and CLIR, especially for languages with strong morphology. **Spelling errors** as well as **spelling variation across languages** are problematic as well. (Ingwersen & Järvelin 2005, 151.) These issues will be discussed more deeply in Chapter 3.3.

2.2 Reductive approaches in IR

Documents include words in their inflected forms, which has an effect on information retrieval. The impact of word inflection on IR varies across languages, however: there are languages with weak morphology (for example English), and on the other hand languages with thousands of word form variants

(for example Hungarian and Hebrew). (Krovetz 1993.) The means to handle morphological variation may be divided into two groups: reductive approaches and generative approaches.

The **reductive approaches** are based on word normalization (see Kettunen 2007). **Word normalization** is a process, which aims to reduce the impact of word inflection. Normalization is performed when documents are indexed. In retrieval, normalized word forms must naturally be used as well (either query word normalization is embedded in retrieval or the user is advised to use normalized query words).

The main normalization approaches are **stemming** and **lemmatization,** which are both language dependent methods. The terminology is not consistent, however. The term normalization is sometimes used to refer to lemmatization. On the other hand, some studies make no distinction between stemming and lemmatization.

2.2.1 Stemming

Stemming is a widely utilized word normalization method in IR. It is a process which aims to reduce the impact of word inflection by mapping inflected word forms into the same stem. Stemming softwares are language specific.

The concept **conflation** may be used instead of stemming. Sometimes conflation is defined to be a broader term than stemming, however. According to Ekmekçioglu, conflation algorithms can be divided into two classes: stemming algorithms and string-similarity algorithms. The latter ones are usually language independent. Thus, conflation may refer to stemming as well as to approximate string matching techniques. (Ekmekçioglu & al. 1996.)

Stemming may be viewed, as well as a way for word normalization, from two other perspectives. It may be thought to be a kind of query expansion mechanism, similar to a thesaurus; or it may be viewed as a clustering process (Krovetz 1993).

The simplest stemming algorithms only remove plural endings. These kinds of algorithms are suitable for languages with weak inflection, for example English. In languages with stronger inflection, suffixes are often joined to a stem one after another. The more advanced stemmers are able to recognize and remove multiple different endings, and some of them apply a multi-step approach, like the Porter stemmer does. These algorithms are *iterative*: they attempt to remove all the joined suffixes. *Longest-match* algorithms remove the longest matching suffix, if there are several possibilities. Longest-match algorithms are easier to program than iterative algorithms, but they require a large dictionary. (Lennon & al. 1981; Krovetz 1993.)

Stemming algorithms may be context-free or context-sensitive. Context-sensitive algorithms use various restrictions for removing suffixes. The restrictions may concern for example the length of the resulting stem. Context-

free algorithms are naturally much easier to develop than context-sensitive algorithms. (Lennon & al. 1981.)

Stemming errors may be classified into three groups: under-stemming, over-stemming and mis-stemming. **Under-stemming** means that the stemmer removes too short a suffix. For example removing only the suffix *s* from the word *babies* would be under-stemming. **Over-stemming** is the opposite thing: removing too much, for example stemming the word *politics* into a stem *poli*. **Mis-stemming happens** when the stemmer takes off a part from the word, which looks like a suffix but is not a suffix. For example removing the suffix *ly* from an English word *cheaply* is often right, but it should not be removed from the word *reply*. (Porter 1981.)

Retrieval research results achieved by various stemmers differ from each other. Kraaij made in 1996 an overview of research results on utilizing stemmers in IR. There are many factors having an impact on the result, for example using linguistic or non-linguistic stemmers, the language, the query length and the document length. Kraaij made experiments with different stemming methods in a Dutch document collection including newspaper articles. He used 36 queries created by test persons. Kraaij found that inflectional stemming was the most successful "simple" linguistic stemming method, and that compound analysis yielded the best results. (Kraaij 1996.)

The impact of stemming on the retrieval result is language dependent. Hollink and colleagues (2004) compared stemmed retrieval results with the inflected result in several European languages. The increase was highest in Finnish (30%), and lowest in Dutch and French (1.2%). (Hollink & al. 2004.)

2.2.2 Lemmatization

The aim of lemmatization is to remove inflectional endings of a given word and to return the basic form, the lemma. Lemmatization softwares are language specific, and they are based on a lexicon as well as morphological analysis of words.

Lemmatizers are often able to decompose compounds. This might be beneficial for languages rich with compounds (for example Finnish, Swedish and German), because it makes headwords of compounds retrievable. Decomposition is based on a lexicon, and thus it is usually not possible for stemmers. There are other decomposition approaches than lemmatization, however: for example, morphological segmentation can be based on unsupervised learning and corpora of various sizes (Goldsmith 2001).

Lexicon dependency can be seen as a drawback of lemmatization: lexicons are always incomplete, because languages are constantly evolving structures, creating new concepts and generating new words for expressing them. In addition, no lexicon can include an exhaustive list of proper names. Thus, there will always be words which lemmatizers are not capable of handling.

Inflectional homonymy is another possible problem for lemmatization. In the case of inflectional homonymy, the lemmatizer will give two or more lemmas for a given word. For example, the English lemmatizer will give two lemmas, *see* and *a saw*, for an input word *saw*. Some of these problems might be solved by part-of-speech tagging. (Pirkola 1999, 49.) Inflectional homonymy is a problem for stemming as well as for lemmatization. In addition, stemmers supply only one stem for each word, and the right sense might be totally lost. For example, a stemmer might give only a stem *saw* for the input word *saw*.

Sometimes lemmatizers give word forms which are correct word forms as such, but which are not real constituents of the current word. Those words can be called **parasite words**. The phenomenon is usually connected with compounds with ambiguous sub word boundaries. (Alkula 2000, 101-102.) The Finnish lemmatizer FINTWOL gives the following interpretations for the word *kahviansa* (partitive case of *her coffee*):

"kahvi#ansa"
"kahvi"

The second interpretation is correct: *kahvi* (coffee), but the first also contains a parasite word *ansa* (*a trap*).

The morphological analyzers utilized in the present research apply the **morphological two-level model**. The model was a result of a project called "Computational analysis of Finnish". One of the aims of the project was to find out which properties a language model should have in order to be capable of coping with morphologically rich languages. (Karlsson 1985, iii.)

The morphological two-level model is on the one hand a **theory** and a formalism describing a word formation (inflection, derivation, compounding, etc.), and on the other hand a concrete **computer implementation** for word-form analysis and synthesis. The formalism has two major components: a **lexicon system** and a collection of **rules**. The lexicon system includes the words of the language as well as all the possible affixes. The rules define how affixes may be joined to words. The morphological two level model is language independent: a new language may be introduced to the program implemented along with the model by describing the lexicon and rules of the language. (Koskenniemi 1983, 9-10.)

2.3 Generative approaches in IR

Normalization is not always applied in indexing. For example many Internet indexes are non-normalized (inflected) indexes. When retrieving in an inflected word form index, the user should add all (or at least the most essential) inflected word forms into the query. Word form generation methods are useful in this:

given a word in its basic form, they aim to generate inflected word forms. Thus, the aim of generative approaches is opposite to the aim of reductive approaches.

One possible method for retrieval in an inflected index is called **FCG (Frequent Case Generation)**. The method is based on a software generating inflected word forms when given a lemma. (See Kettunen & Airio 2006.) Here, only nouns and adjectives are taken into account, because verbs are not very important in IR (Baeza-Yates & Ribeiro-Neto 1999, 169-170).

FCG is based on the idea of identifying the most substantial word forms from the IR point of view: as few word forms as possible, which are enough for facilitating a good retrieval result. The substantial word forms vary from a language to the other. Thus, FCG must be tuned for each language separately. Tuning proceeds in the following way. First, the most frequent case forms are searched through corpus analysis. The required distribution information can be detected in quite a small corpus. Second, retrieval is performed using various combinations of the most frequent case forms (for nouns and adjectives). The outcomes are compared with the best available result, which is usually lemmatization (a lemmatized index and lemmatized queries accordingly), or stemming. Third, the word form combinations yielding the best results are used for the FCG method for the language. (Kettunen & al. 2007.)

For example, tuning FCG for Finnish produced two alternative word form sets: the first including 9 inflected forms and the second 12. When the lemma of the Finnish word *talo* (house) is given to the FCG process producing 9 inflected forms, it will return the following: *talo, talot, talon, talojen, taloa, taloja, talossa, talosta* and *taloon*. Apparently, in an inflected index, a query containing all those forms performs better than a query containing only the lemma.

FCG has been found to be applicable for several languages with varying morphological complexity: English, Finnish, Swedish, German and Russian (Kettunen & al. 2007; Kettunen 2008).

2.4 Approximate string matching

The goal of approximate string matching (string matching that allows errors) is to perform string matching between texts which have suffered some kind of corruption: for example recovering original signals after transmission over noisy channels, interpreting text produced by optical character recognition (OCR) or searching text with typing or spelling errors. The aim of approximate string matching is to find a text where a given pattern occurs, allowing some variation. The idea is based on computing a distance between two strings: if the distance is small enough, the strings are likely variants of each other. (Navarro 2001.)

Research of approximate string matching began in the sixties. The main motivation came from computational biology, signal processing and text retrieval, which are still the largest application areas. (Navarro 2001.)

Approximate string matching techniques involve phonetic coding, edit distance (also called Levenshtein distance) and string similarity based matching. In the first one, a phonetic code is assigned to each string. Two strings are judged to be similar, if they share the same code. Phonetic coding is beneficial in personal name matching because names might sound similar, even if their spelling differs. (Zobel & Dart 1995.) In edit distance, the basic concept is the distance between two strings, which means minimal sequence of operations to transform one string to another. The edit distance operations are the following: insertion, deletion, substitution and transposition (swapping the letters). In string similarity based matching, strings are decomposed into substrings, and the degree of similarity is calculated according to the number of similar substrings. Widely used string similarity metrics are the Jaccard coefficient and the Dice coefficient (Järvelin & Järvelin 2008).

Approximate string matching can also be based on transformation rules. Pirkola and colleagues developed in 2006 an approximate string matching technique, transformation rule based translation (TRT), for identifying translation equivalents in CLIR. A transformation rule spesifies how characters are transformed between a source and a target language. Frequency and confidence factors are important threshold values utilized in TRT rules. (Pirkola & al. 2006.)

We shall introduce two approximate string matching methods, n-gramming and s-gramming more detailed below, because they are the techniques utilized in the present study.

2.4.1 N-gramming

N-gramming is one of the most popular approximate string matching techniques in IR and CLIR. An n-gram is a substring of length *n* of the original string. In applications designed for processing English, the most common types of n-grams have been digrams (n=2) and trigrams (n=3). It is possible to use padding spaces when generating n-grams. Padding spaces mean that space characters in the beginning and in the end of the original string are taken into account. The purpose of padding spaces is to ensure that all the characters of the original string will be equally represented by the n-grams as well. Digrams for the word *substring*, when applying padding spaces ('_'), are the following: *_s, su, ub, bs, st, tr, ri, in, ng,* and *g_*. (Robertson & Willet 1998.)

N-gramming may be applied to the following tasks: spelling error correction, finding spelling variants, spelling error detection, query expansion, name matching, historical text searching, document clustering, and text-signature searching. In spelling error correction, the system assumes that the input word given by a user is spelt incorrectly when it does not occur in the predefined dictionary or corpus. The system identifies the most similar dictionary (or corpus) words and displays candidates to the user. In spelling error detection, a probability score based on n-grams of a string to be checked and n-grams of

dictionary words is calculated: according to the score, the checked string can be labelled as a possibly misspelt word. N-gram based query expansion differs from spelling correction in that many possible variants instead of one are catered. (Robertson & Willet 1998.)

Variation in the structure of proper names, both within a single language and across languages, is quite usual. Often for example geographical names have differing spelling in various languages (*Brussels* vs. *Bryssel*). Also, there is variation for example between American English and British English (*behavior* vs. *behaviour*). N-gramming is useful in solving these kinds of problems, both in monolingual and in cross-lingual retrieval. (Pirkola 1999, 50-51.) For example, if we are performing monolingual retrieval in a lemmatized index, and encounter a query word which the lemmatizer does not recognize, we might benefit from n-gramming: the best matching string(s) can be identified among the target index strings. In CLIR, n-gramming may be applied to words not included in the translation dictionary: they are possibly proper names with differing spelling across languages.

N-gramming has also been applied in various other ways in information retrieval. For example, in the HAIRCUT system of McNamee and colleagues, 6-grams were utilized both in indexing and retrieval. The authors' methods proved to be applicable both for language-independent monolingual retrieval and cross-language retrieval (query translation). (McNamee & al. 2000.)

2.4.2 S-gramming

S-gramming is a modification of n-gramming. The difference is that in s-gramming, digrams are formulated both of adjacent and non-adjacent characters of a string, while n-gramming takes into account only adjacent characters. *S* refers to the number of skipped characters (0, 1, 2, ..., m-2, where *m* refers to the number of characters in the string). The **character combination index** (CCI) indicates the number of skipped characters when s-digrams are formed. Each number in the notation refers to the number of characters between the constituent characters of s-digrams. For example, CCI={{1, 2}} refers to s-digrams formulated by skipping one character and by skipping two characters. CCI={{0}} refers to conventional digrams. Similarly as with n-gramming, padding spaces in the beginning and in the end of the string may be taken into account. (Pirkola & al. 2002.) For example, if CCI = {{0}{1,2}} with padding spaces, the s-grams classes for the string *dogs* are {_d, do, og, gs, s_} and {_o, dg, os, g_, _g, ds, o_}.

S-gramming may be utilized as a translation method for closely related languages. In 2006, Järvelin and colleagues used s-gramming for Norwegian-Swedish translation. They utilized skipgrams with CCI = {{0}{1}} and CCI = {{0}{1,2}}. (Järvelin & al. 2006.) In 2008, Järvelin and Järvelin performed extensive s-gram tests with eleven language pairs. The authors tested seven proximity measures for classified n-grams. They found that the binary proximity

measures (Jaccard coefficient, binary cosine similarity and Hamming distance) gave better results than non-binary (Tanimoto coefficient, cosine similarity and L1 distance), but the differences were mainly due to padding utilized with s-gramming: the binary and non-binary measures performed almost equally when no padding was utilized (Järvelin & Järvelin 2008).

3 Cross-language information retrieval

Cross-language information retrieval is based on translation – either queries are translated into the document language(s), or document(s) are translated into the query language. The latter alternative would be comfortable for the user, but it is expensive and hard to implement. The query translation approach is more common in CLIR, and it is applied in the present research as well. There are three main approaches in CLIR: a dictionary based approach, a corpus based approach and a machine translation based approach (Gachot & al. 2000).

The multilingual task based on query translation is more complicated than the bilingual one, because there are several target languages. An MLIR task may be performed in three alternative ways: 1) the original queries are translated into all the target languages, and monolingual retrieval is performed separately in several target indexes (one index for each language) or in a multilingual index, and finally the result **lists are merged**; 2) the multilingual document collection is translated into the source language and indexed into a single index, after which an MLIR task turns into a monolingual IR task; or 3) the original queries are translated into the target languages (as in the first approach), and then translated queries are merged into a multilingual query, after which retrieval is performed in a multilingual index (all the collections indexed into a single index). (Chen & Gey 2004.) Chapter 3.4 introduces various result list merging approaches.

3.1 Corpus-based CLIR

The corpus-based approach utilizes parallel or comparable corpora. The parallel corpora consist of a collection of pairs of documents in two languages which are translations of each other. Document alignment (sentence alignment, segment alignment, word alignment), which means finding relations between a pair of parallel documents, is a crucial part of the corpus-based approach. (Yang & Kar Li 2004.) There are two main approaches for sentence alignment: length-based and text-based alignment. The former approach is based on the total number of words or characters in a sentence, while the latter utilizes lexical information of sentences. Sentence alignment is based on the assumption of one-to-one translation of sentences. If the number of sentences differs between parallel documents, it is possible to perform segment alignment before sentence alignment. Segment alignment takes into account insertion and deletion of

paragraphs or sentences. Word alignment can be performed in sentence-aligned corpora. (Fung & McKeown 1997.)

Parallel corpora have been utilized in CLIR in various ways, including cross-language pseudo relevance feedback, creation of a cross-language similarity thesaurus, latent semantic indexing and query translation (Dumais & al. 1996; Talvensaari 2008a, 29-30). Parallel corpora have also been used together with other approaches (machine translation, dictionary-based translation) to reduce translation ambiguity. (Oard 1997.)

It is hard to obtain extensive parallel corpora. Alternatively, the corpus-based approach can be applied utilizing comparable corpora. Documents in comparable corpora are not direct translation of each other. Instead, document alignment is based on the similarity between the topics of the documents. (Oard 1997.) Talvensaari and colleagues presented in 2006 their approach to automatically create a comparable document collection for CLIR. The source language in their tests was Finnish, and the target language was English. The document collections contained newspaper articles. (Talvensaari & al. 2006.) In 2007, Talvensaari and colleagues continued developing their system. The source language in these tests was Swedish, while the target language was English. They created a similarity thesaurus for query translation. (Talvensaari & al. 2007.)

There are some factors which affect the quality of the parallel / comparable corpora approach in a CLIR task. First, the corpora must be large enough, and they must include rare words as well. Second, the domain of corpora must fit the topic of the queries. Third, the quality of the alignments naturally affects the quality of the approach: parallel corpora are better than noisy comparable corpora. Thus, comparable corpora are best as complementary translation resources. (Talvensaari 2008b.) On the other hand, a domain specific comparable corpus can even outperform a high-quality parallel corpus with more general vocabulary (Talvensaari & al. 2008). Although the parallel corpora approach often performs quite well, it is difficult and expensive to carry out, because it includes collecting large bilingual corpora and dividing the sentences (and possibly paragraphs as well) into fragments. (Yamabana & al. 2000.)

3.2 Machine translation -based CLIR

Machine translation (MT) systems analyze the source text, including morphological, syntactic and semantic analysis utilizing special lexicons. The aim of machine translation is to translate complete sentences, and it is the only translation approach applicable for document translation. MT systems return only one translation variant for a word, which may cause loss of recall in retrieval (Yamabana & al. 2000). In addition, MT-based query translation may not produce very good results with short source queries which are typically not complete sentences and thus do not provide sufficient contextual information for translation (Chen & Gey 2004; Kishida 2005).

Despite the possible drawbacks mentioned above, MT-based query translation has performed quite well in IR tests, when the MT system has been of good quality, and source queries have been complete descriptions of information needs, e.g. TREC topics (see Oard 1998; Rosemblat & al. 2003; Huang & al. 2007). On the other hand, the performance of a poorer MT system can be boosted by combining other methods with translation, for example pseudo relevance feedback. It is also possible to combine translations of two or more MT systems in order to achieve a better query. (See Jones & Lam-Adesina 2002; Chen & Gey 2004.)

Document translation would be beneficial for users of a retrieval system, but translating a large document collection into numerous languages is exorbitant. Fujii and Ishikawa proposed in 2000 a lighter version of document translation: only retrieved documents were translated (see Fujii & Ishikawa 2000).

3.3 Dictionary-based CLIR

The dictionary-based approach[1] relies on standard machine-readable bi- or multilingual dictionaries. In dictionary-based CLIR, each query word is translated into the target language. The translation process produces none, one or more translation equivalents for each source word. (Hedlund 2003, 26-27.) Because all translation variants are included, there is no fear of losing the right one (supposing that the dictionary is good enough), which might happen in machine translation. There is even the possibility that translation acts like a query expansion, because translation dictionaries often include synonyms. On the other hand, there is also a possibility of retrieving noise in the case of an ambiguous source word. The dictionary-based approach is the most common CLIR approach, because translation dictionaries are often relatively cheap and easy to use.

3.3.1 A dictionary-based CLIR task compared with a monolingual IR task

A simple dictionary-based CLIR task consists of three subtasks: 1) source word translation, 2) information retrieval, and possibly 3) result list merging (see Figure 2). If source words are not in their basic form, they should be lemmatized before translating. This is vital, because translation dictionaries do not contain words in inflected forms. Lemmatized source query words are translated into the target language. After translation, reductive (normalization, stemming) or generative (FCG) methods may be applied, depending on the type of the target

[1] *With dictionary-based CLIR, the query translation approach is denoted here and later.*

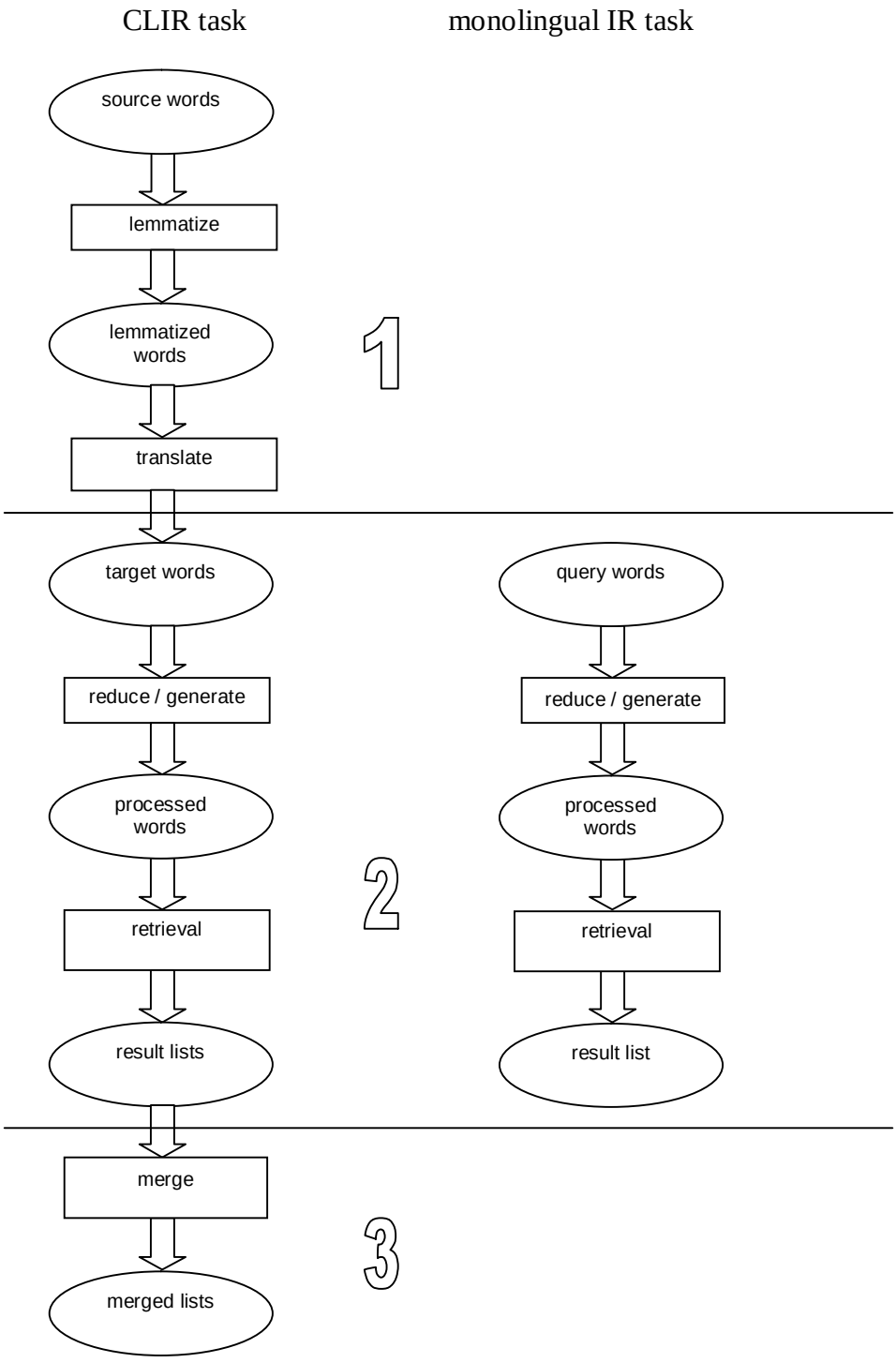

Figure 2. *A simple dictionary-based CLIR task and a monolingual IR task*

index. The question of OOV words is an issue for lemmatization and translation. Depending on the system, the OOV words might be forwarded as such to the next phase, or for example handled utilizing approximate string matching techniques (to keep Figure 2 simple, this is omitted). The next phase of the process is the actual retrieval. In the case of a bilingual task, as well as a multilingual task with a shared index and merged queries, retrieval produces one result list, and subtask 3) is omitted. If the approach of separate indexes is followed in a multilingual task, the separate result lists are merged according to the selected merging strategy.

The simplest monolingual IR task means just retrieval with the given query. Often reductive or generative methods are applied to query words before retrieval, depending on the index type. Figure 2 shows the relation between a simple dictionary-based CLIR task and a traditional monolingual IR task. Both share the subtask 2). More accurately, a CLIR task consists of three quite separate functions: translation, traditional IR task and result list merging.

3.3.2 Linguistic problems and their solutions in dictionary-based CLIR

The main problems of dictionary-based CLIR may be summarized as follows: 1) **untranslatable search keys**, 2) **compounds,** 3) **phrases,** 4) **word inflection**, 5) **lexical ambiguity** (Pirkola & al. 2001).

Untranslatable search keys

The problem of untranslatable words is unavoidable, because there cannot exist a dictionary which would list all the possible words of a language. There are three main reasons for this. First, languages are evolving systems, and dictionary updating is not fast enough. In addition, in some languages, for example Finnish, it is possible to create new compounds by combining words. Including all potential word combinations in a dictionary is not possible. Second, translation dictionaries do not include personal names and most other proper names. Third, general dictionaries do not usually include special terms, which may be primary search keys in some queries. (Pirkola & al. 2001; Hedlund 2003, 27-28.) Also machine translation systems share the problem of untranslatable words.

An untranslatable word may be a proper name, a geographical name or a special term not included in the dictionary. In many CLIR systems, untranslatable words are included as such into the target query. In some cases this is a successful method, and in other cases not. Sometimes spelling of an untranslatable word is only slightly different in the source language compared to the target language. N-gram based matching or some other approximate string matching technique, or transliteration, is the solution for the spelling problem. In n-gramming, the untranslatable word is compared to the target index words in

order to find the most similar variants. (Pirkola & al. 2001; Kishida 2005; Palmer & Ostendorf 2005.)

Compounds

Untranslatable compounds cause problems for CLIR. The problem may be solved by decompounding before translation. Whether this is a good approach, depends on the type of the compound: is it compositional or non-compositional. The meaning of a compositional compound can be derived out of its parts (for example the Finnish compound *kaupunginhallitus*, *city government*). A non-compositional compound is a word whose meaning is not directly derivable from its parts. An example of this could be the English compound *strawberry*. Thus, when parts of a non-compositional compound are translated separately, the result may be nonsense, while separate translation of parts suits well for compositional compounds. (Pirkola & al. 2001; Chen & Gey 2004.)

Stemmers are usually not capable of decompounding, but lemmatizers are. Decompounding is not always an adequate action, because the components of a compound may not be in their basic form. For example, the first part of a compound may be in genitive (the Finnish compound *kaupunginhallitus*, where *kaupungin* is genitive of the word *kaupunki*, *city's*). The parts of the compound should be lemmatized as well, and translated after that (Hedlund 2002). In many Germanic languages (German, Dutch and the Scandinavian languages) there is a specific feature of using joining morphemes in compounds. The German compound *Handelsvertrag* (*trade agreement*) is a good example of this: the first part of the compound is *Handel* (*business, trade*) and the latter is *Vertrag* (*contract, agreement*). The parts are connected by a joining morpheme *s*. (Hedlund 2003, 22; Hollink & al. 2004.) Joining morphemes pose challenges for lemmatizers.

If parts of a compound included in a source query are translated separately, they might probably be re-joined when constructing the target language query. Pirkola and colleagues suggested in 2001 that each translation equivalent of the first part of a compound should be joined by the proximity operator with each translation equivalent of the latter part, generating all the combinations. (The proximity operator tells that words should occur within a narrow window in a document.) The proximity statements should be joined using the synonym operator. (The statements joined by the synonym operator are treated as the instances of the same statement.) The statement should finally be enveloped by the *sum* operator of the INQUERY retrieval system (the system computes an average weight of query key weights). (Pirkola & al. 2001.) For example, the following query would be created when translating the German word *Handelsvertrag* into English:

> SUM(SYN(PROX (business contract) PROX(business agreement) PROX(trade contract) PROX (trade agreement)))

Here, SYN stands for the synonym operator, PROX for the proximity operator and SUM for *sum* operator. In 2003, Pirkola and colleagues tested the performance of the proximity operator with compounds in Finnish-English retrieval. They found that the proximity operator was not beneficial. A better result was achieved by simply enclosing translation equivalents of each source word with the synonym operator in the following way: SYN(business trade) SYN(contract agreement). (Pirkola & al. 2003.)

Phrases

Phrases cause problems for CLIR in languages where phrases are utilized rather than compounds to create multi-word expressions (e.g. English). When the source query includes a phrase which is present in the dictionary, the correct translation may be lost without phrase recognition. Phrases may be divided into two groups, compositional and non-compositional, similar to compounds. An example of a compositional phrase could be *information retrieval*, and that of a non-compositional a *hot dog*. (Pirkola & al. 2001; Kishida 2005.)

Well translated phrases have a positive effect on a retrieval result, while poorly translated phrases may impair the result. Translation accuracy may be more important for phrases than for single words. (Ballesteros & Croft 1997.) On the other hand, it is possible to make a distinction between recognition of compositional and non-compositional phrases. Pirkola and colleagues suggested in 2001 that recognition of compositional phrases is often not vital, because right translations will be achieved by translating parts of a phrase separately, while recognition of non-compositional phrases is more important. (Pirkola & al. 2001.)

Word inflection

The most common solution for the **word inflection** problem is normalization, which is described in Chapter 2.2. In dictionary-based CLIR, a stemmer can be used in source word normalization only if stemming is applied to the dictionary as well, because the head words in the dictionaries are lemmas. Dictionary stemming might add translation ambiguity because of over-, under- and mis-stemming. After translation, either a stemmer or a lemmatizer may be utilized, depending on the target index. Lemmatizers rely on lexicons. Thus, like translation dictionaries, lemmatizers suffer from the OOV problem. If a source query word cannot be lemmatized, it is probable that it cannot be translated either. N-gramming techniques can be utilized to handle unidentified words. Stemmers do not have these kinds of problems, because their function is often mainly based on rules.

In the case of an inflected word form index, it is possible to apply for example approximate string matching techniques or inflected word form generation for handling the translated words. The approximate string matching approach can be applied similar to that used with untranslatable words described above, but here it is applied to translatable as well as untranslatable words. Word form generation requires a language dependent software, which takes a lemma as input, and which gives required inflected word forms for the given lemma as output (see Chapter 2.3).

Lexical ambiguity

Lexically ambiguous words, like homonyms, usually cause no problems in human communication, because the right denotation and connotation may be concluded from the situation and the sentence context. Machine translation systems try to do the same automatically, but translation dictionaries alone are not capable of this: they give all translation variants. (Pirkola & al. 2001; Hedlund 2003, 18-19; Kishida 2005.)

Lexical ambiguity in CLIR may be reduced by part-of-speech tagging, corpus-based disambiguation or query structuring. In part-of-speech tagging, only translation equivalents having the same part-of-speech with the source language word are selected. Corpus-based disambiguation methods involve query expansion to reduce the effects of bad translation equivalents, the use of word co-occurrence statistics for selecting the best translations, and selection of translation equivalents on the basis of aligned sentences. If the retrieval system includes a synonym operator, query structuring is a way to avoid problems caused by lexical ambiguity in CLIR. A synonym-operator is used in the query to denote that the words grouped with the operator should be treated as expressions of one word, and given an equal query weight. (Pirkola & al. 2001; Kishida 2005.)

3.4 Result list merging

In the traditional multilingual information retrieval approach, there is a separate index for each target language. Retrieval is performed separately in each index, and finally the result lists are merged. Thus, the process consists of multiple bilingual runs and result list merging. There are several possible ways to merge the result list.

1) *The round robin approach.* An item from each result list is taken by turn. The approach is based on the idea that document scores are not comparable across the collections. If one is ignorant about the distribution of the relevant documents in the retrieved lists, it is

reasonable to assume the distribution to be symmetric. (Hiemstra & al. 2001.)

2) *The raw score approach*. Result lists are merged according to the scores. This approach is based on the assumption that document scores are comparable across collections. (Hiemstra & al. 2001.)

3) *The normalized score approach*. If document scores are not comparable between the collections, it might be reasonable to normalize them. Normalization may be performed by dividing each score by the maximum score reached in the collection. A variant of this is to divide each score by the difference between the maximum and the minimum document score values:

   $C' = (C - C_{min}) / (C_{max} - C_{min})$

   where C is the original document score, C_{min} the minimum score reached in the collection and C_{max} the maximum score. (Powell & al. 2000.)

4) *The weighted score approach*. Weights can be based upon document score and / or the collection ranking information. If the collections are assumed to be different, the collection score might be used in weight calculation. The collection score is based on the idea of a collection retrieval inference network (CORI): the query is first used to retrieve a ranked list of collections, and collection scores are given based on this list.

$$s' = s \times \left(1 + |C| * \left(c - K\right) / K\right)$$

   Where

   s' = the weighted score
   s = the document score,
   |C| = the number of collections searched,
   c = the collection score, and
   K = the mean of the collection scores

   (Callan & al. 1995.)

In addition, we present here two approaches which we have developed for Study II.

5) *The dataset size based approach*: The approach is based on the assumption that it is likely that more relevant documents are found in a large dataset than in a small dataset. The number of documents taken from single result sets is calculated as follows:

   $T * n / N$

   where T is the number of documents per topic in the single result list, n is the dataset size and N is the total number of documents (the sum of documents in all the collections).

6) The score difference based approach: Every score is compared with the best score of the topic. Only documents with the difference of scores under the predefined value are taken to the final merged list. This is based on the assumption that documents whose scores are much lower than the score of the top document, may not be relevant.

In recent years, there have been attempts to develop new, better merging approaches. Martínez-Santiago and colleagues introduced in 2006 2-step RSV (Retrieval Status Value) for result list merging. In the method, retrieved documents are re-indexed according to their query vocabulary. This requires query vocabulary alignment: original query words must be aligned with their translations. For queries with partial word-level alignment, the authors developed four mixed models. The authors found that their approach outperformed the traditional approaches. The traditional approaches reach about 65-70% of the theoretical optimal performance, while 2-step RSV reached about 85-90%. (Martínez-Santiago & al. 2006.)

4 Approaches to IR evaluation

Various approaches have been developed to facilitate IR evaluation. In the present chapter, we are going to present some of them.

4.1 Relevance

Relevance is the basic concept for all the IR performance measures. The concept can be traced back to the beginning of the early fifties and the first retrieval systems. In those days, relevance was treated as a synonym of relatedness of documents and queries. The first debates of the concept of relevance emerged in the mid-fifties. *User relevance* and *relevance to a subject* were then introduced. The first formal studies on relevance were published in the sixties. (Saracevic 1970, 114-116.)

There are various manifestations of relevance in the modern IR research. *Algorithmic* (or system) *relevance* means similarity of documents with a query in the logical or statistical sense. *Topical* (or subject) *relevance* is a relation between the topic of a query and the topic of documents. *Pertinence* (or cognitive relevance) takes into account the user's information need: pertinence means cognitive correspondence, informativeness and novelty of documents for the user. (Schamber & al. 1990; Schamber 1994; Saracevic 1996; Cosijn & Ingwersen 2000; Saracevic 2007.) *Situational relevance* (or utility) means usefulness of documents in the perceived situation, perceived by the user himself, not by others or objectively. It is also relevance in relation to an individual's stock of information, and it changes as the stock changes. Situational relevance can be introduced by inductive logic as the relationship between an item of information and an individual's situation. (Wilson 1973; Schamber 1994; Saracevic 2007.) *Psychological relevance* is based on the idea that users prefer documents which will have an effect on their current cognitive state. Thus, a document might be psychologically relevant for a user even if it would not directly handle the topic of the query he posed. (Harter 1992; Saracevic 2007.)

Relevance in IR research usually means topical relevance. The aim is to assess relevance of documents for a topic as neutrally as possible. The shift from laboratory tests towards user tests has brought new approaches to the concept of relevance: relevance is seen to resemble more pertinence or situational relevance than an objective measure. (Barry 1994.)

Relevance was originally assessed using a binary scale: either a document is relevant to a topic or it is not relevant. Beginning from the nineties, there have been opinions that the binary scale is not adequate for a realistic evaluation: some documents may be very useful and contain a lot of information about the topic, while the others just mention something related to the topic. It is obvious that users generally prefer highly relevant documents over marginally relevant ones, which justifies the use of a graded relevance scale. (See Sormunen 1994; Borlund 2000; Järvelin & Kekäläinen 2000; Voorhees 2001; Järvelin & Kekäläinen 2002; Sormunen 2002.) Despite the benefits of graded relevance, binary relevance still dominates in many test collections, for example TREC (The Text Retrieval Conference)[1] and CLEF (Cross Language Evaluation Forum)[2]. On the other hand, NTCIR (NII Test Collection for IR Systems) Project[3] has utilized graded relevance for its collections since the 7th workshop in 2007, and graded relevance was also used in TREC-9 in 2000.

Sormunen (2002) created a test collection with graded relevance assessments from a TREC dataset. The collection was created by reassessing TREC documents which originally had binary assessments. Sormunen found that only 16% of documents originally assessed as relevant were highly relevant in the graded scale. Thus, utilizing the non-binary scale might alter the IR evaluation results. (Sormunen 2002.)

The graded relevance scale is more often adopted in user tests than in laboratory tests. In this thesis also, the graded scale is adopted in the user tests, while the test collections used in the laboratory tests have binary assessments.

4.2 Laboratory oriented IR research

The aim of **laboratory oriented IR research** is to test whether a new system or a new technique performs better than an old one, or to compare two or more existing systems or techniques with each other. In laboratory oriented IR research experiments, there are neither any test persons, nor queries formulated by them. Instead, user queries are **topics** created by a research group or a researcher (for example in TREC). Topics are descriptions of single user information needs, and they are worded by native language speakers. In laboratory tests, users' relevance assessments are compensated by **relevance corpora**. Relevance corpora are composed in the following way. First, candidate documents are retrieved for each topic, utilizing multiple variant queries, retrieval systems and matching methods. Each query yields a result list. Next, a document pool is built by picking documents for each topic from the beginning

[1] *http://trec.nist.gov/*

[2] *http://www.clef-campaign.org/*

[3] *http://research.nii.ac.jp/ntcir/*

of the result lists. The number of selected documents per a list is decided beforehand. Finally, relevance assessors evaluate each document in the pool.

The traditional IR laboratory model has been criticised for its lack of realism. Some of the problems of the laboratory model may be summed in the following way. First, real users and their situational tasks are completely ignored, and users' relevance assessments are substituted by a single assessor's (necessarily biased) assessments. Second, laboratory tests are mostly batch tests: there is no real interaction. Third, laboratory tests are based on static test collections, which are topically narrow, while real collections are more diverse. Fourth, laboratory tests assume that documents are independent and neglect document overlap. Fifth, test collections are structurally simple, while some real-life collections have an interesting internal structure. (Kekäläinen & Järvelin 2002b; Ingwersen & Järvelin 2005, 8-9.)

4.3 User oriented research

Recent literature on information seeking and retrieval recognizes IR as only one means of information access: a user's task, its phase, situation, context, IR strategies and many other factors have an impact on it (Ingwersen & Järvelin 2005, 1-3). A great deal of IR research is still based on laboratory tests, however. In some tests, users are only partially involved. For example, in the interactive track of TREC, users interact with a system, but document relevance is based on previous assessments (see Over 1997).

Thus, everything is controlled in IR laboratory experiments. When users are involved, many problems connected with laboratory research experiments may be solved, but new problems arise. First, users introduce interactivity to retrieval; this research approach is called **interactive information retrieval (IIR)**. According to Robins, there is not any dominant IIR model. Instead, there are several models which try to describe the dynamics of interaction, for example Saracevic's stratified model of interactive IR (see Saracevic 1997); Belkin's episodic model of IR interaction (see Belkin 1996); Spink's interactive feedback and search process model (see Spink 1997); and Ingwersen's global model of polyrepresentation (see Ingwersen 1996). The field of interactive IR is so wide that it offers challenges to interdisciplinary collaboration, including information science, psychology, business and computer science. (Robins 2000.)

There are more variables to test in user tests than in laboratory tests: the impact of interfaces / document representation on user satisfaction / retrieval performance, as well as user interaction with the system. These things also introduce challenges, because some kind of control over them is needed: without control it is not possible to make any kind of comparisons. In addition, users' personal features affect retrieval, and they can be controlled only partially (by utilizing as homogenous groups as possible or by taking users' personal differences into account when interpreting results). If users are allowed to create

their own search tasks, new problems arise. The tasks may be quite different in nature, some are easy to carry out, and the others are very difficult. Also the number of relevant documents may vary, as well as users' conception of relevance. Thus, comparison of retrieval effectiveness across tasks may be difficult, but that is not always the only aim of IR research experiments. Nevertheless, measurement and finding causes and effects in user-oriented research settings is challenging.

4.3.1 Simulated work task approach

As stated above, realistic evaluation of interactive information retrieval is difficult. In order to facilitate it, Borlund (2002) introduced an IIR evaluation package. The components of the package are the following: involvement of potential users as test persons; using individual and potentially dynamic information need interpretations; and applying multidimensional and dynamic relevance judgements. When possible users are involved, the individual and dynamic nature of information needs can be taken into account. The use of individual and dynamic information need interpretations is also called the **simulated work task approach**, where information need is seen as a consequence of a problematic situation. Dynamic relevance assessments are incorporated by having relevance judged in relation to the need interpretation. Multidimensionality means that relevance may be perceived differently by different users: it is present when there are several users assessing relevance. (Borlund 2000, 27 and 78-79.)

The concept of a work task is central in IR, and therefore it is potentially useful for IIR evaluation. The simulated work task approach is an attempt to make the work task operable. It is based on short cover-stories describing a situation leading to retrieval. Cover-stories are semantically quite open descriptions of a given work task situation. Test persons are advised to formulate a query (or many queries) based on the cover-story. The purpose of the simulated work task situation is two-fold. First, it triggers a simulated information need, which leads to individual information need interpretations. Second, it serves as a platform against which situational relevance is judged. (Borlund 2000, 80-84.)

Multidimensional and dynamic relevance refers to users' subjective decisions about document relevance with respect to criteria, degree and time. Situational relevance is employed, because it is the most comprehensive and embracing type of subjective relevance. Besides situational relevance, other relevance types can be applied as well, but they restrict the subjectivity of assessments. (Borlund 2000, 84-89.)

4.4 Recall and precision

Recall means the ability of a retrieval system to uncover relevant documents. Recall alone is not enough when measuring the effectiveness of a retrieval system, however, because the retrieval result usually contains also unwanted documents which bring noise. *Precision* means the ability of a retrieval system to uncover *only* relevant documents. (Lancaster 1968; Hull 1993.) More precisely:

precision = r / n
recall = r / N
where r = number of relevant document retrieved
N = total number of relevant documents
n = number of documents retrieved
(Hull 1993; Baeza-Yates & Ribeiro-Neto 1999, 75.)

The formulas above are for calculating precision and recall for collections with binary relevance assessments. Kekäläinen and Järvelin have developed measures where graded relevance assessments can be utilized instead of binary. Generalized precision gP is computed in the following way:

$$gP = \sum_{d \in R} r(d) / n$$

Correspondingly, generalized recall gR is computed like this:

$$gR = \sum_{d \in R} r(d) / \sum_{d \in D} r(d)$$

where R is a set of n documents retrieved from a database D = $\{d_1,\ldots,d_N\}$ and r(d) are relevance scores of documents, ranging from 0.0 to 1.0. The same calculations can be applied to these averaged measures as for the traditional ones: e.g. averages over queries, precision averages over recall levels, and drawing performance curves. (Kekäläinen & Järvelin 2002a.)

Sometimes 10%, 20% or 100% of relevant documents beginning from the start of the result list are considered. These points are called standard recall levels, the number of them being usually 11 (0%, 10%,…,100%). The relationship between recall and precision on standard recall levels is often illustrated as a curve. It is possible to draw such a curve for a single query, but it is usually drawn to illustrate the average performance of several queries.

The average precision at the recall level *r* is calculated in the following way:

$$\overline{P}(r) = \sum_{i=1}^{m} \frac{P_i(r)}{m}$$

where $\overline{P}(r)$ = the average precision at the recall level r

m = the number of queries

$P_i(r)$ = the precision at the recall level r for the i_{th} query

(Baeza-Yates & Ribeiro-Neto 1999, 76-77.)

It is also possible to average the precision values (usually 11 values) in order to have a single figure over the performance quality. The disadvantage is that the values are interpolated and thus less reliable. The average non-interpolated precision can be used instead in order to avoid the problem. It is called *mean average precision* (MAP), and it is calculated in the following way:

$$MAP = \frac{1}{m}\sum_{j=1}^{m}\frac{1}{N_j}\sum_{i=1}^{C} r_i \frac{\sum_{s=1}^{i} r_s}{i}$$

where m = the total number of queries

N_j = the total number of relevant documents for query j

C = the cut-off rank

r_i = relevance of a document at the rank i (0 or 1)

(based on Kraaij 2004, 88.)

Each query has the same weight when calculating the MAP. Thus, MAP is quite sensitive to topics with only a few relevant documents. Despite this, MAP is commonly used in IR evaluation, including the studies of the present thesis.

4.5 Cumulated gain

Precision and recall are based on assumption that the document position in the result list is insignificant and that each document is either relevant or irrelevant. Cumulated gain questions these assumptions. It is based on two ideas: 1) highly relevant documents are more valuable than marginally relevant documents, and 2) the position of a document is important: the user examines only the top documents (see Järvelin & Kekäläinen 2000; Järvelin & Kekäläinen 2002). When calculating cumulated gain, the ranked document lists are turned to gained value lists by replacing document ids by the corresponding relevance score.

The direct cumulated gain vector CG is calculated recursively in the following way:

$$CG[i] = \begin{cases} G[1], & if\ i = 1 \\ CG[i-1] + G[i], & otherwise \end{cases}$$

where G[i] denotes the position *i* in the gain vector G.

It is obvious that the ranked position of a document has resonance for a user: the greater the position of a relevant document, the less valuable it is, because it is less likely that the user will examine the document. Thus, the greater the rank,

the smaller share of the document value should be added to the cumulated gain. A discounting function progressively reduces the document value as the rank increases. The cumulated gain vector with discount DCG is defined in the following way:

$$DCG[i] = \begin{cases} CG[i], & if\ i < b \\ CG[i-1] + G[i]/({}^{b}\log i), & if\ i \geq b \end{cases}$$

The logarithm-based discount cannot be used at rank 1, because ${}^{n}\log 1 = 0$ (Järvelin & Kekäläinen 2002).

Various versions of the cumulate gain formulas have been developed among IR researchers (see for example Tang & al. 2006; Wu & Crestani 2008; Carterette & Jones 2008).

The cumulated gain formulas model the user persistence: smaller logarithm values model impatient users and bigger persistent users (Järvelin & Kekäläinen 2002).

4.6 Statistical tests

It is often important to measure whether possible differences in evaluation scoring are statistically significant. The null hypothesis is that the methods being tested are equal: there are only minor differences between their performance. The idea of significance tests is to find out whether differences could have occurred by chance. (Hull 1993.)

The t-test, the paired Wilcoxon signed-rank test and the sign test are the possible alternatives for statistical tests when there are two methods to be compared (Hull 1993). Smucker and colleagues found in 2007, however, that the nonparametric Wilcoxon and the sign test have a poor ability to detect significance and may also lead to false detections of significance (Smucker & al. 2007).

The two-way analysis of variance (ANOVA) is the parametric statistical approach for comparing more than two methods. There are many non-parametric versions of ANOVA. These tests assume that the query effect and the effect of the evaluation methods are independent and additive. The Friedman test, a generalization of the sign test, is one of the nonparametric ANOVA versions. (Hull 1993.)

Statistical tests are applied in the present thesis in the following way. The Wilcoxon signed-rank test was applied in Study III. The t-test was utilized in Study IV. In Study V, we used general linear model (repeated measures).

5 Summary of the studies

This section presents a summary of the studies of this thesis. Section 5.1 presents the tests settings: the query translation system, NLP resources, retrieval systems and test collections utilized in the studies. Section 5.2 introduces Studies I, II and III, which address the impact of various normalization methods and translation dictionaries on monolingual, bilingual and multilingual IR. Study IV is presented in Section 5.3. It discusses approximate string matching and the FCG method in bilingual IR. Section 5.4 deals with CLIR user tests and presents Study V.

5.1 Test settings

This chapter describes tests settings and resources utilized in the studies of the current thesis. The resources are summarized in Table 1.

Table 1. *Resources used in the studies*

Resources	Study I	Study II	Study III	Study IV	Study V
Dictionary-based translation systems	UTACLIR, C-version	UTACLIR, C-version	UTACLIR, C-version	UTACLIR, Java-version	UTACLIR, C-version
Translation dictionaries	GlobalDix Motcom English-Finnish	GlobalDix	GlobalDix	GlobalDix	GlobalDix Motcom Finnish-Swedish
MT systems					Babelfish
Lemmatizers	ENGTWOL FINTWOL	ENGTWOL FINTWOL GERTWOL SWETWOL	ENGTWOL FINTWOL GERTWOL SWETWOL	ENGTWOL FINTWOL GERTWOL SWETWOL	ENGTWOL FINTWOL GERTWOL SWETWOL
Stemmers		Dutch English Finnish French German Italian Spanish Swedish	English Finnish German Swedish		
Word form generators				Finnish: FGEN Swedish: Grim	
Retrieval system	INQUERY	INQUERY	INQUERY	Lemur Indri	Google

5.1.1 Dictionary-based query translation

Query translation is one of the main themes of the present thesis. The query translation system utilized in the studies is UTACLIR, developed at the University of Tampere. The idea of UTACLIR is to translate a given source language query into the target language, processing each query word individually. UTACLIR is just a framework for the process: it utilizes external language resources for translation, word normalization, stop-word removal and approximate string matching.

The UTACLIR system tries to overcome most of the problems present in a CLIR task: problems caused by compounds, word inflection and untranslatable words. Processing of an individual source word may be described as follows (see Figure 3). First the source word is normalized with a lemmatizer. Stemmers are not suitable for this phase, because dictionary head words are given as lemmas. The lemmatizer produces a lemma (or multiple optional lemmas) of the source word, if the word is recognized. If the lemmatizer does not recognize the source word, it is sent as such to the next phase, which is stop-word removal. If the word turns out to be a stop-word, processing ends here. In the other case translation is attempted. If no translation variants exist, s-gram techniques are adopted to find the best matching variants from the target index. If there are translation variants, they are normalized with either a stemmer of a lemmatizer, depending on the target index, and target stop-words are removed. Last, target words derived from a single source word are structured into a synonym clause.

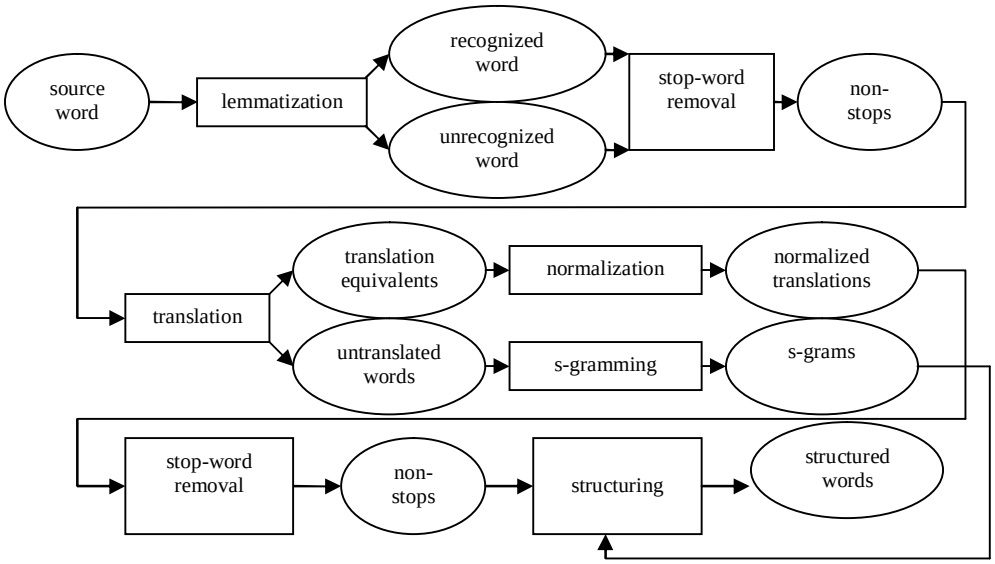

Figure 3. *An overview of processing a word with the UTACLIR system.*

The UTACLIR system was developed for the CLEF campaigns[1] at the University of Tampere (UTA). The original UTACLIR system was evaluated and improved in the CLEF 2000, 2001 and 2002 campaigns. It consisted of separate programs for each language pair (Finnish-English, German-English and Swedish-English). UTA started to develop a new UTACLIR version connected with the EU project CLARITY[2] in 2001. In this new version of UTACLIR, there are no separate programs for distinct language pairs, but a single program is able to handle all the languages requested. The source and the target language of the runs as well as possible optional language resources utilized are expressed by special codes given as input.

UTACLIR consists of library archives containing general and resource specific functions. General functions perform tasks, which are common for all the language pairs, and call the language-specific functions. Language pairs and resources may be added easily by adding pertinent language-specific functions, which satisfy the function prototype definitions. (Hedlund & al. 2002.) UTACLIR was originally programmed in C, and later in Java. The Java version was utilized in Study IV, and the C version in the other four studies.

5.1.2 NLP resources

Various language resources have been utilized in the studies included in the present thesis: translation dictionaries, machine translation systems, lemmatizers, stemmers, word form generators and stop-word lists.

The following translation dictionaries have been utilized via the UTACLIR system:

- Motcom GlobalDix multilingual translation dictionary (18 languages, total number of words 665 000, an average 36944 / language) by Kielikone plc. Finland
- Motcom English-Finnish translation dictionary (110 000 entries) by Kielikone plc. Finland
- Motcom Finnish-Swedish translation dictionary (84 000 entries) by Kielikone plc. Finland

We utilized Motcom GlobalDix in all of the papers included in this thesis. The Motcom English-Finnish translation dictionary was utilized in Study I and the Motcom Finnish-Swedish translation dictionary in Study V.

In addition to dictionary-based translation with UTACLIR, machine translation was utilized in Study V. The MT system utilized was

- the publicly available MT system Babelfish http://babelfish.altavista.com/

In the translation phase of CLIR, topic words must be lemmas, because translation dictionary head words are lemmas. A lemmatizer is thus needed for

[1] *Cross Language Evaluation Forum, http://clef.iei.pi.cnr.it:2002/*

[2] *Information Society Technologies Programme, IST-2000-25310*

the source language. Retrieval is performed with the translated words. Before retrieval in a normalized index, translated words must be normalized accordingly. The following lemmatizers and stemmers were utilized:

- Lemmatizers FINTWOL, SWETWOL, GERTWOL and ENGTWOL by Lingsoft plc. Finland
- Stemmers for Spanish and French, by ZPrise
- A stemmer for Italian, by the University of Neuchatel
- A stemmer for Dutch, by the University of Utrecht
- Stemmers for English, German, Finnish and Swedish, SNOWBALL stemmers by Dr Martin Porter

The Finnish, Swedish and German lemmatizers (FINTWOL, SWETWOL and GERTWOL) are capable of splitting compounds, while the English lemmatizer, ENGTWOL, is not. Compounds are rare in English: thus, English decompounding is only seldom needed.

For retrieval in a non-normalized index, some other tools, for example word form generators, may be utilized. The word form generators utilized in the tests are the following:

- Finnish word form generator FGEN from Teemapoint
- Swedish word form generator Grim from Numerical Analysis and Computer Science of Royal Institute of Technology, Sweden

Stop-word removal is also an essential phase of CLIR, and thus stop-word lists are essential resources. The stop-word lists utilized in the tests are the following:

- English stop-word list (429 stop-words), created on the basis of INQUERY's default stop-word list for English
- Finnish stop-word list (773 stop-words), created on the basis of the English stop-word list
- German stop-word list (1318 stop-words), created on the basis of the English stop-word list
- Swedish stop-word list (499 stop-words), created at the University of Tampere

Lemmatizers and stop-word lists were utilized in all the five studies, and stemmers in Studies II and III. The word form generators were used in Study IV (see Table 1).

5.1.3 Information retrieval systems

We utilized three retrieval systems in our research. The INQUERY retrieval system was used in Studies I, II and III. Lemur Indri was utilized in Study IV and Google in Study V (see Table 1).

INQUERY

The indexing and retrieval system *INQUERY* was created by the Center for Intelligent Information Retrieval at the University of Massachusetts. INQUERY is based on a probabilistic retrieval model called the inference network. (Callan & al. 1992.)

The inference network model emphasizes retrieval based on combination of evidence. It is based on *data fusion*: different text presentations (for example words or phrases) as well as various versions of the query (for example natural language and Boolean) can be combined in a consistent probabilistic framework. (Broglio & al. 1994.)

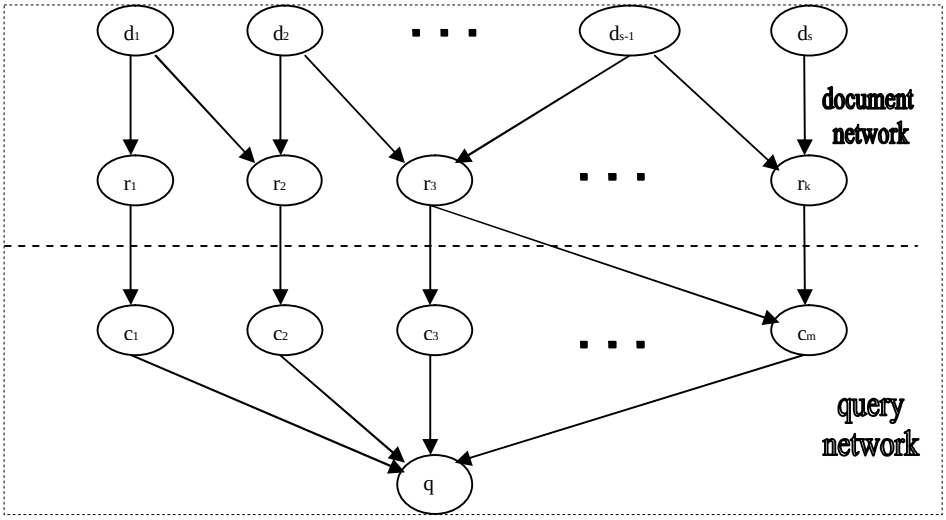

Figure 4. *A simple document retrieval inference network (Callan & al. 1992).*

INQUERY creates networks for documents and queries. In Figure 4, there is a simple document network with two levels of abstraction: the document text level *d* and the content representation level *r*. The query network consists of a query level *q* and the concept level *c*. A document node d_i represents the proportion that a document satisfies a user query, and a content representation node r_k the proportion that a concept has been observed. A query node *q* represents the proportion that an information need is met, and a concept node c_j the proportion that a concept is observed in a document. Document nodes and a query node are assigned the value *true*, while content representation nodes may be *true* or *false*. The value of an arc between d_i and a content representation node r_k is the conditional probability $P(r_k \mid d_i)$, and the value of an arc between r_k and a query concept node c_j is the belief in the proposition. (Callan & al. 1992.)

The document network is created when indexing a database by mapping documents with nodes, and storing the nodes in an inverted file. INQUERY supports various indexing techniques, for example word-based indexing and phrase indexing. It is possible to attach a stemmer or a lemmatizer with the

indexing software. Indexing produces inverted files that allow efficient performance also with large databases. INQUERY operators are used when formulating queries: it is possible to define new concepts and how to calculate the belief in those concepts. INQUERY query processing is multi-phased, and many of the steps are identical with the indexing steps. The evaluation process involves probabilistic inference, and it utilizes the inverted files and the query represented as an inference net. (Callan & al. 1992; Broglio & al. 1994.)

INQUERY calculates the belief in term *t* within document *d* in the following way:

$$w_{td} = 0.4 + 0.6 * \frac{tf_{td}}{tf_{td} + 0.5 + 1.5 * \frac{length(d)}{avglen}} * \frac{\log \frac{N + 0.5}{n_t}}{\log(N + 1)}$$

where n_t = the number of documents containing term *t*
N = number of documents in the collection
avglen = the average length (in words) of documents in the collection
length(d) = the length (in words) of document *d*
tf_{td} = the number of times term *t* occurs in document *d*

(Allan & al. 1997.)

The following operators of the INQUERY system are used in the tests:

- Sum Operator #sum (T1 ...Tn): The terms or the nodes contained in the sum operator are treated as having equal influence on the final result. The belief values provided by the arguments of the sum are averaged to produce the belief value of the #sum node.
- Un-ordered Window Operator #uwN(T1 ... Tn): The terms contained in a uwn operator must be found in any order within a window of N words in order for this operator to contribute to the belief value of the document.
- Synonym Operator #syn(T1 ... Tn): The terms of the operator are treated as instances of the same term[1].

The weight for the synonym clause is calculated in the following way:

$$0.4 + 0.6 * \frac{\sum_{t \in S} tf_{td}}{\sum_{t \in S} tf_{td} + 0.5 + 1.5 * \frac{length(d)}{avglen}} * \frac{\log \frac{N + 0.5}{n_s}}{\log (N + 1.0)}$$

where tf_{td} = the number of times term *t* occurs in document *d*
S = a set of search words within the syn operator

[1] *INQUERY Query Help*. Copyright by the CIIR and/or Sovereign Hill Software, http://ciir.cs.umass.edu/irdemo/inqinfo/inqueryhelp.html

n_s = the number of documents containing at least one key of the set S

N = number of documents in the collection

avglen = the average length (in words) of documents in the collection

length(d) = the length (in words) of document *d*

(Kekäläinen & Järvelin 1998.)

In effect, the synonym operator reweights the synonym set at the search stage, treating all the words in the set as occurrences of the same word.

Google

Google is an Internet search engine, which finds its documents for indexing by crawling the Web. Google has some features different from a traditional search engine. It makes use of the link structure of the Web to calculate a quality ranking. Google also utilizes link graph to improve search results. PageRank, the link analysis algorithm used by Google, counts inlinks to a given page and normalizes the sum by the number of outlinks in the page. (Brin & Page 1998.)

According to Brin and Page (1998), PageRank *PR* of a page *A* is calculated in the following iterative way:

$$PR(A) = (1-d) + d * \sum_{i=1}^{n} \frac{PR(T_i)}{C(T_i)}$$

where d = a damping factor between 0 and 1

T_i = a page pointing to the page A

n = number of pages pointing to the page A

$C(T_i)$ = the number of outgoing links of the page T_i

(Brin & Page 1998.)

Google developers do not describe the actual implementation of their current algorithm, however. According to an techinical overview in Google homepages (2009), the principles of PageRank are still similar to that described above, but the formula might be more complicated: it includes over 500 million variables and over two million terms. (See Google 2009.)

Thus, the PageRanks form a probability distribution over indexed web pages, and the sum of the PageRanks of those pages will be one. A high PageRank may be either due to the high number of pages pointing to a page or due to high PageRank of the pointing pages or both. In the first case, the page is probably of quality because it is widely cited, and in the second case, citations (or maybe only one citation) are of very high quality. (Brin & Page 1998.)

PageRank can be said to simulate user behaviour. PageRank is the probability that a random user visits a page, given a number of random starting web pages and clicking links until she gets bored. The damping factor *d* is the probability of the user getting bored. (Brin & Page 1998.)

Google offers Java interfaces (Google API) for its users. Utilizing them, it is possible to integrate Google with one's own retrieval interface. This kind of solution was created for the tests described in Study V.

Lemur Indri

The Lemur Toolkit was developed in collaboration between the Computer Science Department at the University of Massachusetts and the School of Computer Science at Carnegie Mellon University. The Lemur toolkit facilitates research in language modelling and information retrieval. Lemur supports many research applications such as ad-hoc retrieval, site-search, and text mining. (Lemur 2008.) The Lemur system is written in the C and C++ languages, and is designed for Unix operating systems.

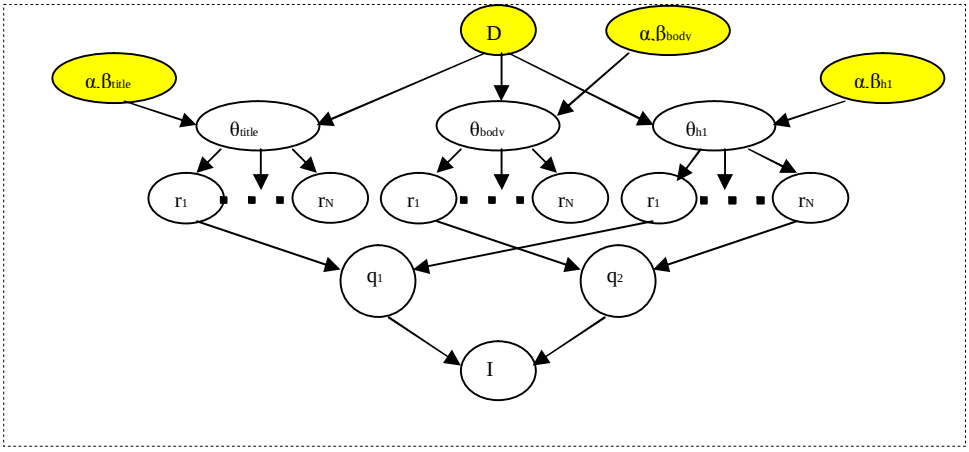

Figure 5. *Indri's inference network retrieval model (Abdul-Jaleel & al. 2004).*

Lemur Indri is the newest search engine from the Lemur project. Indri's retrieval model combines the language modelling and the inference network approaches to information retrieval (see Figure 5). A document is represented as a sequence of tokens in a language modelling framework. A multinomial language model over the vocabulary is then estimated based on this sequence. Often documents are represented as multisets of binary feature vectors (θ_{title}, θ_{body} and θ_{h1} in Figure 5). A feature node *r* corresponds to document features that can be represented in an Indri structured query. A query node operator *q* is a soft probabilistic operator. (Abdul-Jaleel & al. 2004.)

The model allows structured queries similar to INQUERY queries, but evaluation is performed utilizing language modelling estimates. In the inference network framework, documents are ranked according to $P(I \mid D, \alpha, \beta)$, the belief the information need I is met given document D and hyperparameters α and β as evidence. (Abdul-Jaleel & al. 2004.)

Indri creates various data structures in the indexing phase: a compressed inverted file for the corpus, including term position information; compressed inverted extent lists for each field indexed in the corpus; a vector representation of each document; and a random-access compressed version of the corpus text. Indri facilitates interfacing a lemmatization or a stemming tool with the indexing software. (Abdul-Jaleel & al. 2004.)

Indri supports popular structured query operators from INQUERY. It parses PDF, HTML, XML and TREC documents, and supports UTF-8 encoded text.

5.1.4 Test collections and topics

The Web (October 2006 – February 2007) served as a test bed in Study V, while CLEF test collections were utilized in the other four studies. We utilized the Dutch, English, Finnish, French, German, Italian, Swedish and Spanish test collections of CLEF (see Table 2). All the collections were utilized in Study II. The Finnish collection was used in Study I, while English, Finnish, German and Swedish collections were used in Study III, and Finnish and Swedish in Study IV. The Swedish and the Finnish training collections were utilized in Study IV (see Table 3).

Table 2. *Test collections*

Language	**Collection**	**Size (MB)**	**Number of documents**
Dutch	NRC Handelsblad 1994/95	299	84 121
	Algemeen Dagblad 1994/95	241	106 483
English	Los Angeles Times 1994	425	113 005
	Glasgow Herald 1995	154	56 472
Finnish	Aamulehti late-1994/1995	137	55 344
French	Le Monde 1994	158	44 013
	SDA French 1994	86	43 178
	SDA French 1995	88	42 615
German	Frankfurter Rundschau 1994	320	139 715
	Der Spiegel 1994/95	63	13 979
	SDA German 1994	144	71 677
	SDA German 1995	144	69 438
Italian	La Stampa 1994	193	58 051
	SDA Italian 1994	86	50 527
	SDA Italian 1995	85	48 980
Spanish	EFE 1994	511	215 738
	EFE 1995	577	238 307
Swedish	Tidningarnas Telegrambyrå 1994/1995	352	142 819

Table 3. *Training collections*

Language	Collection	Size (MB)	Number of Documents
Swedish	Göteborgs-Posten and Helsingborgs Dagblad 1994	280	161 336
Finnish	Tutk (Aamulehti, Kauppalehti and Keskisuomalainen 1988-1992)	135	53 893

The topic sets utilized in the studies with CLEF collections were CLEF 2002 and 2003 topic sets. The former includes 50 topics, and the latter 60. The number of topics corresponding to each test collection (i.e. number of topics which have relevant documents in the collection) differs, however, because there are not relevant documents for each topic in each database (see Table 4).

In Study I, CLEF 2002 English and Finnish topic sets were used, and CLEF 2003 English topic set in Study II. In Study III, CLEF 2003 English, Finnish, German and Swedish topic sets were utilized, and in Study IV the same, except the German one. The simulated work task approach was applied in Study V, and the tasks were created at the University of Tampere.

Table 4. *Number of topics per collection for CLEF 2002 and CLEF 2003*

	Number of topics	
CLEF Collection	**CLEF 2002**	**CLEF 2003**
Dutch	50	56
English	42	54
Finnish	30	45
French	50	52
German	50	56
Italian	49	51
Spanish	50	57
Swedish	49	54

5.2 Summary of studies on linguistic and approximate string matching methods in IR and CLIR

Studies I, II and III address the effects of various word normalization and approximate string matching methods as well as translation tools on monolingual, bilingual and multilingual retrieval.

5.2.1 Study I

Research problems

Study I concentrated on monolingual and bilingual retrieval. The monolingual part of the study addressed lemmatized Finnish retrieval (lemmatized query words, the lemmatized index). When lemmatization is utilized, OOV words might cause problems, and we wanted to test the performance of n-gramming on them. Compounds form a remarkable part of the Finnish vocabulary. Parts of the compounds are often content bearing words. Therefore decompounding often improves the retrieval result. We tested various compound handling approaches.

In bilingual retrieval, the impact of the translation dictionary may be substantial. We tested two dictionaries of different levels in bilingual (English-Finnish) retrieval.

Thus, the research problems of Study I were:

1. What is the impact of approximate string matching of OOV words on monolingual (Finnish) retrieval?
2. What is the impact of the operator utilized to envelope parts of compounds on monolingual (Finnish) retrieval?
3. What is the impact of alternative translation dictionaries on bilingual (English-Finnish) retrieval?

Methods

The search engine utilized in Study I was INQUERY (see 5.1.3). The UTACLIR system was utilized for query translation (see 5.1.1). The CLEF Finnish test collection (see Table 2) with the CLEF 2002 Finnish and English topics (see Table 4) formed the test bed. For lemmatization, we utilized FINTWOL and ENGTWOL (see 5.1.2).

In the monolingual test of Study I, the compounds occurring in the topics were decompounded, and the constituents were lemmatized. Finally, the lemmatized constituents of the compound were enveloped with an appropriate operator and added to the query. Four runs were performed, and in two of them, n-gramming was applied to words not recognized by the lemmatizer. The n-gram system returned two strings: one best match among the index lemmas and one best match among non-recognized index words. These strings were enveloped by the INQUERY's synonym operator.

The impact of n-gramming and various proximity operators in enveloping parts of compounds were studied in the following combinations: a) n-gramming for OOV words and the synonym operator for compounds, b) n-gramming for OOV words and the proximity operator uw3 for compounds, c) no n-gramming for OOV words and the synonym operator for compounds, d) no n-gramming for OOV words and the uw3 operator for compounds.

In the bilingual part of Study I, the translation dictionaries which were compared with each other were the Motcom GlobalDix multilingual translation dictionary and the Motcom English-Finnish bilingual translation dictionary, both from the same producer (see 5.1.2). The number of Finnish entries in GlobalDix is quite small. In addition, the translation strings sometimes contain lots of noise, which might have an effect on the results.

Results

The first research question addressed the impact of n-gramming on monolingual Finnish retrieval. We found that handling OOV words with n-gramming did not have a remarkable impact on the result. The second research question dealt with the impact of the operator to envelope parts of compounds. The alternative b), n-gramming for OOV words and the uw3 operator for parts of compounds, achieved the best result: average precision was 35.2%. The second best was d), no n-gramming for OOV words and the uw3 operator for parts of compounds (average precision 32.0%). The third best was the alternative a) (average precision 27.0%), while c) gave the poorest result (24.0%). Thus, the answer to the second research question is that the operator enveloping parts of compounds has an impact on the result: the proximity operator seems to outperform the synonym operator.

The impact of operators to combine parts of compounds has been tested on bilingual retrieval (see Hedlund & al. 2001a; Hedlund & al. 2001b; Pirkola & al. 2003), but not so widely on monolingual retrieval. Pohlmann and Kraaij introduced in 1996 a system which decompounded compounds present in queries and formatted new compounds of the parts. Thus, the system formulated new compounds: it did not combine compound parts with operators. (See Pohlmann & Kraaij 1996.) Monz and De Rijke tested in 2002 decompounding in Dutch, German and Italian retrieval, but they did not utilize any operators to envelope parts of compounds (see Monz & De Rijke 2002). Thus, the approaches for handling compounds in monolingual retrieval presented in Study I have not been tested earlier.

The third research question addressed the impact of the translation dictionary on the result of bilingual retrieval. We noticed that the MOT translation dictionary delivered almost three times more translation variants compared to GlobalDix, which had an impact on the result: the average precision of the English-Finnish run with the GlobalDix translation dictionary was 24.6%, while it was 32.6% with the MOT English-Finnish dictionary. The change between GlobalDix and the MOT bilingual dictionary is +32.5%. Thus we can conclude that the extent and the quality of the translation dictionary have a remarkable impact on the result. The conclusion might be quite self-evident, but the level of the effect is more remarkable than had been expexted.

5.2.2 Study II

Research problems

The theme of **Study II** was multilingual (English -> Dutch, English, Finnish, French, German, Italian, Spanish, Swedish) retrieval. The result list merging approach was applied. There were two things which might affect the retrieval result: the normalization approach and the result list merging approach. The research problem of the Study II was:

4. Which normalization and result list merging approach combination performs best in multilingual (English -> Dutch, English, Finnish, French, German, Italian, Spanish, Swedish) retrieval?

Methods

The search engine utilized in Study II was INQUERY. The UTACLIR system was utilized for translation, and GlobalDix was used as the translation dictionary. The test collections were CLEF Dutch, English, Finnish, French, German, Italian, Spanish and Swedish collections, and the topics were the CLEF 2003 English topics. For lemmatization, we utilized FINTWOL, ENGTWOL, GERTWOL and SWETWOL. The Dutch, English, Finnish, French, German, Italian, Spanish and Swedish stemmers were utilized for stemming. These tools and data sets are introduced in Section 5.1

In Study II, two indexes were built for English, Finnish, German and Swedish: the lemmatized index and the stemmed index. For Dutch, French, Italian and Spanish, only the stemmed index was built (we had no lemmatizers available for those languages).

The tested result list merging methods were the raw score method, the dataset size based method and the score difference based method (see Section 3.4). The merged result lists were produced from the following single result lists: a) lemmatized queries and the lemmatized index (English / Finnish / German / Swedish), stemmed queries and the stemmed index (Dutch / French / Italian / Spanish); and b) stemmed queries and the stemmed index (English / Finnish / German / Swedish / Dutch / French / Italian / Spanish).

Results

Our fourth research question addressed the performance of various word normalization and result list merging approach combinations in multilingual retrieval. The differences between the results of various methods were not remarkable. The index type had more impact on the results than the merging method. All the merges of lists with the type a) achieved better results than the

merges of lists of the type b). The best result, average precision 20.2%, was achieved with the dataset size based merging method, the next was round robin (20.1%), the third was the score difference method (19.9%), and the fourth was the raw score (19.8%) method, all with lists of type a). The corresponding results with lists of type b) varied from 18.5% to 18.7%. The impact of the result list merging approach on the retrieval result was only minor, and the effect of the normalization was not much bigger. In addition, the results of all the multilingual runs were quite poor. The answer to the fourth research question is that there was no remarkable difference between the performance of various word normalization – result list merging approach combinations. Prior research has attained similar results concerning the impact of the merging approach (see Chen 2003; Moulinier & Molina-Salgado 2003; Braschler & al. 2003). On the other hand, word normalization combined with various result list merging approaches has not been investigated earlier.

There are not many systems available utilizing result list merging approaches, but merely documents in various languages are indexed in a merged index. Thus, research should perhaps be directed towards retrieval strategies in a merged index.

5.2.2 Study III

Research problems

Study III dealt with compounds. The compound problem rose accidentally with our CLEF English-Finnish and English-Swedish experiments in 2003 (Study II) when we had a closer look at the differences between the lemmatized run in the decompounded index and the stemmed run. We noticed that phrases among topic words had impact on the differences: queries in the lemmatized decompounded index outperformed queries in the stemmed index. This led us to deeper analyze the issue: do compounds have an impact as well on monolingual retrieval results in various index types, and which other issues might cause performance differences in mono and bilingual retrieval.

Compounds had also been a focus in the previous CLIR research at our University: Hedlund studied processing of compounds in bilingual dictionary-based retrieval in 2002. Compounds occurring in queries were a focus of her work, decompounding them, normalization of components of compounds, translation of components and query structuring for compounds and their components. (See Hedlund 2002.)

For Study III, monolingual (English, Finnish, German and Swedish) and bilingual (English-Finnish, English-German and English-Swedish) retrieval was performed. We wanted to find out the impact of normalization on monolingual retrieval: is it essential to apply normalization or can we survive without it? We

compared various normalization approaches in monolingual and bilingual retrieval. Especially we wanted to test the impact of compounds on the result.

The research problems of Study III were:

5. Which word normalization method gives the best result in monolingual (English, Finnish, German and Swedish) retrieval: stemming, lemmatization without decompounding or lemmatization with decompounding?
6. Which word normalization method gives the best result in bilingual (English-Finnish, English-German and English-Swedish) retrieval: stemming, lemmatization without decompounding or lemmatization with decompounding?

Methods

INQUERY was utilized for indexing and retrieval in Study III. For translation we utilized the UTACLIR system, and GlobalDix was used as the translation dictionary. The test collections were CLEF English, Finnish, German and Swedish collections, and the topics were the CLEF 2003 English, Finnish, German and Swedish topics. For word normalization, we used the English, Finnish, German and Swedish lemmatizers and stemmers. The utilized tools and data sets are described in Section 5.1

For Finnish, German and Swedish, four indexes were built: inflected, stemmed, lemmatized with decompounding and lemmatized without decompounding. Three indexes were built for English: inflected, stemmed and lemmatized without decompounding.

The following monolingual runs were performed: English, Finnish, German and Swedish inflected, stemmed, lemmatized (in the compound index), and in addition lemmatized runs in the decompounded index for Finnish, German and Swedish.

The bilingual runs were the following: English-Finnish, English-German and English-Swedish. For all the language pairs, two lemmatized runs (one in the decompounded index, one in the compound index) and one stemmed run were performed.

Results

All our monolingual non-English normalized runs, except one (the run in the German lemmatized compound index) performed significantly better (Wilcoxon signed rank test at the 0.01 level) than the inflected run. No significant differences could be found between the English inflected run and any of the English normalized runs. The results are in line with the majority of the previous results of English monolingual retrieval. English is morphologically quite a simple language, which explains the good performance of inflected retrieval. On

the other hand, Finnish, German and Swedish are morphologically more complex, and thus inflected retrieval performs more poorly.

Our fifth research question asked which of the various normalization approaches performs best in monolingual retrieval. The inflected run was the baseline. In Finnish, Swedish and German retrieval, the best result was achieved with the lemmatized decompounded index, the next best with the stemmed index, and the worst with the lemmatized compound index. Decompounding had the largest impact on Swedish: the run in the lemmatized compound index performed 19.1% worse than the run in the lemmatized decompounded index. Also differences between the Swedish run in the lemmatized decompounded index and the run in the stemmed index were statistically significant, as well as differences between the German run in the lemmatized decompounded index and the run in the lemmatized compound index. No other statistically significant differences could be found.

Closer query specific analysis revealed some reasons for the performance differences between the runs. First, in some cases, index decompounding acted like document expansion, because also documents including the query word as a constituent of a compound could be found. Thus, some queries performed better in the decompounded index than in the other index types. Second, under-stemming caused problems for the stemmed run. Third, OOV words caused problems for the lemmatized runs. OOV words were added as such to the query, in their inflected word form. Thus documents including the word in some other word form were not retrieved. Approximate string matching techniques could have helped in this. The stemmed run gave better results for many queries including words not recognized by the lemmatizer: the functionality of the stemmers we used is not based on any dictionaries, but is rule-based.

In bilingual IR, phrases included in source language topics might pose a challenge. First, it is a challenge to identify phrases, and second, the translation dictionary might not include the phrases even if they were identified. Among the CLEF 2003 English topics, which were used in this study, there were 42 fixed phrases: thus, they are compounds in a compound-oriented language like Finnish. Out of them only one (fast food) could be translated utilizing GlobalDix. Most of the phrases (35) were rare (for example *diamond industry*), but the rest were frequent phrases (for example *mobile phone*), which should have been included in the dictionary. No phrase recognition was applied in the study. On the other hand, the benefits of phrase recognition would have been scarce, because only one phrase could have been translated.

Our sixth research question addressed various normalization approaches in bilingual retrieval. Retrieval in the lemmatized index with decompounding performed best among all the bilingual runs: the average precision of the English-Finnish run was 35.5%, English-Swedish 27.1% and English-German 31.0% (see Table 5). Among English-Finnish and English-German runs, the next best was the run in the lemmatized compound index. In English-German, the run in the lemmatized compound index (26.4%) performed almost equally with the run in the stemmed index (25.7%), while in English-Finnish, the stemmed run

(20.8%) performed clearly worse than the run in the lemmatized compound index (29.0%). In English-Swedish, the result of the stemmed run (19.0%) was slightly better than that of the run in the lemmatized compound index (17.4%).

Table 5. *Non-interpolated average precision of bilingual runs (source language English) for all relevant documents averaged over queries.*

Target language	Index type	Average precision %	Diff. % (from the baseline)	Change % (from the baseline)	Diff. % (from the lemm. non-decomp.)	Change % (from the lemm. non-decomp.)
Finnish	lemmatized + decompounded	35.5				
Finnish	lemmatized, non-decompounded	29.0	-6.5	-18.3		
Finnish	Stemmed	20.8	-14.7	-41.4	-8.2	-28.3
Swedish	lemmatized + decompounded	27.1				
Swedish	lemmatized, non-decompounded	17.4	-9.7	-35.8		
Swedish	Stemmed	19.0	-8.1	-29.9	1.6	9.2
German	lemmatized + decompounded	31.0				
German	lemmatized, non-decompounded	26.4	-4.6	-14.8		
German	Stemmed	25.7	-5.3	-17.1	-0.7	-2.7

The reason for the better performance of the lemmatized decompounded index compared with the lemmatized compound index or the stemmed index lies in the structure of the source and the target languages. In English, phrases are utilized to express multiple part concepts, while Finnish, Swedish and German are compound oriented languages. When an English query included a phrase, the parts were translated separately. On the other hand, target documents included a compound to express the corresponding phrase. The compound did not match the parts of a phrase. Decompounding split the compound into its constituents, which matched the parts of the phrase.

Individual queries were analyzed in order to detect the reasons for the performance differences between the various approaches. The first reason detected was the impact of compounds: parts of the phrases included in the topics were translated separately, producing separate translations, while the corresponding target language expression was a compound. Only the lemmatized decompounded index included the parts of compounds. Thus, queries including phrases performed better in the lemmatized decompounded index than in the other index types. The second and the third reason for the performance differences were connected with stemming: especially the Finnish stemmer behaved inconsistently (over-stemming and under-stemming).

Stemming is the most popular normalization method in IR research, and English is the most popular retrieval language, but research on stemming of non-English retrieval is increasing (see Popovič & Willet 1992; Larkey & al. 2002; Hollink & al. 2004; Kettunen 2004). Instead, decompounding is not a very

widely investigated approach. In 2004 Braschler and Ripplinger studied stemming and decompounding in German monolingual retrieval (see Braschler & Ripplinger 2004). Alkula (2000) investigated monolingual Finnish retrieval in lemmatized compound index and lemmatized decompounded index (see Alkula 2000). None of the preceding studies cover comparison of stemming, lemmatization without decompounding, and lemmatization with decompounding, which are included in Study III. In addition, Study III covers comparison of four languages (English, Finnish, German and Swedish).

5.3 Summary of the study IV on linguistic and approximate string matching methods in bilingual inflected retrieval

5.3.1 Study IV

Research problems

Dictionary-based translation mostly produces target queries with lemmas. Some other cases may appear as well: thus, it is appropriate to lemmatize or to stem translations accordingly with the target index. This approach is possible if the target index is stemmed or lemmatized. It is not possible to build a stemmed or a lemmatized index, however, if there are no normalization tools available for the target language, Also, some operational indexes (for example many Web indexes) are non-normalized. Retrieval in a non-normalized index with a query including only lemmas may produce a poor result. This might be dependent on the morphology of the target language: English as the target language behaves differently from Finnish.

The idea for **Study IV** arose from the existing approaches and resources: Kettunen and colleagues had developed the FCG method and tested it in monolingual retrieval, achieving promising results. The next obvious phase was to test FCG in bilingual retrieval. Bilingual retrieval poses extra challenges for FCG: multiple translations (relevant and irrelevant synonyms) for each query word, possible compounds and phrases, OOV words and cross-language spelling variants. A natural point of comparison for FCG was s-gramming, an approximate string matching technique developed at University of Tampere, which has been shown to perform well in CLIR experiments.

The aim of the Study IV was to test approximate string matching techniques and generative approaches for CLIR in a non-normalized index. We hypothesized that s-gramming performs better with untranslatable words than the generative approach. We decided to test also the combined method of s-gramming and the generative approach.

The research questions are:

7. Which of the following inflected methods performs best in bilingual retrieval (English-Finnish, Swedish-Finnish, English-Swedish, Finnish-Swedish) in a non-normalized index: approximate string matching methods or generative methods, or combination of those?
8. What are the reasons for performance differences between various methods in distinct language pairs?
9. Is the result based on the best inflected method commensurate with that of the gold standard?

Methods

The search engine used in Study IV was Lemur Indri (see 5.1.3), and the language pairs were English-Finnish, Swedish-Finnish, English-Swedish and Finnish-Swedish. The test collections were the CLEF Finnish and Swedish collections (see Table 2). The Göteborgs-Posten and Helsingborgs Dagblad collection was utilized as the Swedish training collection and Tutk as the Finnish training collection (see Table 3). We needed the training collections for selecting the numbers of target index words for the s-gram runs. In order to do this we performed runs in the training collections before the actual runs for all our language pairs. We tested the performance of queries consisting of various rates of target index words (from one to twelve). In English-Finnish and Swedish-Finnish training runs, the best performance was achieved with queries including twelve index words, and the second best with queries including eleven index words. The corresponding rates in English-Swedish and Finnish-Swedish were seven and ten.

Altogether six runs were performed in the non-normalized index for each language pair: the raw translation run (the baseline), two s-gram runs (with various number of target index words selected by s-gramming), two FCG runs (with various numbers of inflected word forms), and the combined run (s-gramming and FCG).

Results

Our seventh research question addressed the performance of various methods in inflected bilingual retrieval. Not surprisingly, the baseline, retrieval with raw translations performed worst among the inflected runs in all the language pairs (see Table 6). The average precision of the English-Finnish run was 11.2% and that of the English-Swedish run 18.1%. For Swedish-Finnish the average precision of the inflected run was 11.7% and for Finnish-Swedish 14.3%. Inflection of the Finnish language was the reason for the poorer performance of the English-Finnish and Swedish-Finnish baseline runs: the queries included only one word norm, the one given by the dictionary - usually singular

nominative, which covers fewer than 30% of Finnish noun occurrences, on the average.

Table 6. *Mean average precision (%) of the runs*

Language pairs	Raw transl.	Fi-sgram_11 Sv-sgram_7	Fi-sgram_12 Sv-sgram_10	Fi-FCG_9 Sv-FCG_2	Fi-FCG_12 Sv-FCG_4	Comb-Ined	Lemma-tized
English-Finnish	11.2	31.2	31.0	32.4	32.5	**33.3**	39.0
English-Swedish	18.1	25.3	23.8	25.1	27.3	**27.4**	34.2
Swedish-Finnish	11.7	27.2	26.6	28.0	27.9	**28.1**	36.2
Finnish-Swedish	14.3	**27.7**	27.5	22.6	23.9	27.4	35.5

The combined run outperformed the other runs in English-Finnish, English-Swedish and Swedish-Finnish, but only slightly. In Finnish-Swedish, the s-gram run with eleven target index words selected by s-gramming performed best.

The eighth research question addressed the reasons for performance differences in distinct language pairs. The reasons were detected by query-based analysis. The performance of queries of the better performing s-gram run was compared with that of the better performing FCG run, and the queries with differences equal or more than 10% were counted. The number of queries where FCG outperformed s-gramming was quite even between all the language pairs (it was nine in English-Finnish and English-Swedish, and eleven in Swedish-Finnish and Finnish-Swedish). On the other hand, the number of queries where s-gramming performed better was even with all the other language pairs (five or seven) except Finnish-Swedish, where it was twelve. Finnish-Swedish was the only language pair where the s-gram runs clearly outperformed the FCG runs. There were two main reasons for this: spelling variation between languages and source word inflection.

The ninth research question dealt with the gold standard in bilingual dictionary-based retrieval: we wanted to know, whether the best inflected run is commensurate with it. The lemmatized run outperformed the other runs in all the language pairs (see Table 6). The mean average precision was 39.0% in the English-Finnish lemmatized run. Thus, the combined run performed 14.6% worse than the lemmatized run. The mean average precision of the English-Swedish lemmatized run was 34.2%: the combined run gave 19.9% worse result. In Swedish-Finnish, the combined run performed 22.4% worse than the lemmatized run, and in Finnish-Swedish 22.8% worse (see Table 6). In Swedish-Finnish, English-Swedish and Finnish-Swedish, the differences are large and also statistically significant (t-test, $p < 0.05$), but in English-Finnish, the combined run achieved a statistically comparable result with the lemmatized run. Thus, the best retrieval result in the inflected index is commensurate with the result of the lemmatized run only in English-Finnish.

Some IR indexes, for example many Web indexes, are non-normalized. Also, there are languages lacking appropriate lemmatization / stemming tools. Thus, there is a need for a practical method to perform bilingual retrieval in an inflected word form index. There is not much research on the issue, however.

Thus, Study IV introduces new information as well as new approaches to bilingual retrieval in a non-normalized index.

5.4 Summary of the study V on linguistic methods in interactive CLIR

5.4.1 Study V

Research problems

The traditional IR and CLIR research has been based on laboratory tests, which have several benefits (see 4.2). Laboratory tests do not show the actual benefit of a system for a potential user, however. In addition, topics used in laboratory tests have been formulated by native language speakers, who presumably might need no query translation at all. It is obvious that monolingual queries always achieve better results than translated queries because of multiple reasons, including the following: translation increases ambiguity, the quality of the translation dictionary might not be perfect, OOV words as well as compounds / phrases cause difficulties. **Study V** rose from the observation that the benefit of bilingual retrieval for real users has not been tested widely. In addition, most of the interactive CLIR experiments have been performed with test collections. Thus, there is not much research on the applicability of query translation with large collections or in connection with Web engines. The performance of translated queries compared with target language queries in Web retrieval has not been tested widely, either. In addition to this, we wanted to clarify whether users' language skills, topic familiarity or the topic vocabulary familiarity (in the target language) had any impact on the result. Thus, the research questions were:

10. Which perform better, dictionary-based translations (Finnish-Swedish, English-German, Finnish-French) or the user formulated target language queries (Swedish, German, French)?
11. To what degree do the following user characteristics affect the IR performance achieved through the translated queries / the target language queries: the language skills, the topic domain familiarity, the topic vocabulary familiarity?

Methods

The search engine utilized in the Study V was Google (see 5.1.3). The Web served as a test bed. The language pairs in the tests were Finnish-Swedish, English-German and Finnish-French. UTACLIR was utilized for dictionary-

based query translation, and the translation dictionaries we used were MOT Finnish-Swedish dictionary and MOT GlobalDix by Kielikone plc. For English-German, also machine translation with Babelfish was utilized (see 5.1.2).

Test participants were hired among students at the University of Tampere. There were twelve participants in the Finnish-French and in the English-German test, and eighteen in the Finnish-Swedish test. Each participant selected four search tasks from ten optional tasks: we wanted to give participants the possibility to select tasks according to their interests. The aim was to compare the performance of the translated queries with that of the target language queries. Thus, participants were asked to formulate a source language query and a target language query for each task. The source language queries were translated into the target language. Retrieval was performed with both queries: the user formulated target language query and the UTACLIR translated query (and the Babelfish translated query in English-German). The result lists retrieved with these queries were merged before forwarding the result to participants. The participants were asked to evaluate each document as not relevant, marginally relevant, fairly relevant or relevant according to their subjective opinion.

We wanted to measure the impact of language skills on the results. Thus, we calculated a language skills measure utilizing high school report grades, high-school graduation grades and active / passive language usage. We divided the participants into two language skills groups: 1- moderate skills and 2 - good skills.

Our aim was also to gauge the impact of topic familiarity / the target language (topic) vocabulary familiarity on the results. We asked participants to fill in a questionnaire concerning the topic after performing each task. Thus, we had two measures for the tasks: a topic familiarity measure and a vocabulary familiarity measure, which both had three values: 1 - not at all familiar, 2 - a little familiar, 3 - very familiar.

Results

Our tenth research question asked which performed better, dictionary-based translations or the user formulated target language queries. In Finnish-Swedish, the average generalized precision of the target queries was 21.8%, while it was 24.5% for the UTACLIR queries. In the English-German test, the average generalized precision of the target queries was 26.1%, for the Babelfish queries 29.0% and for the UTACLIR queries 32.5%. In the Finnish-French test, the average generalized precision of the target queries was 33.7% and that of the UTACLIR queries was 16.9%.

The users were divided into groups according to their language skills, the topic familiarity and the topic vocabulary familiarity. The eleventh research question addressed the impact of those characteristics on the results. In the Finnish-Swedish test, two of the between-subject factors were significant: the language skills and the vocabulary familiarity. When the participants were

classified into two groups according to their language skills, the average generalized precision of the UTACLIR queries in the groups was almost the same (23.5% and 25.7%). The average generalized precision of the target queries for those in the group with moderate skills was 12.8%, while it was 33.5% for the group with good skills. Thus, for those who belonged to the group with moderate skills, query translation was beneficial, while it did not help the participants in the group with good skills. The vocabulary familiarity seemed to correlate both with the results of the target queries and the UTACLIR queries: the more familiar the vocabulary, the better results. The groups were not even, however: the group which was very familiar with the vocabulary included only five query pairs. Thus, there may be some impact of coincidental factors.

In the English-German test, only the topic familiarity had a significant effect on the performance differences between the target / UTACLIR / Babelfish queries. The participants not at all familiar with the topic got better results with the translated queries than with the target query, which is reasonable. Those who were a little familiar got as good results with all the queries. The results of the participants very familiar with the topic are confusing: they got poor results with the target queries. Some target queries in this group were defective and some included spelling errors, which explains the result. Topic familiarity had a significant effect on the performance differences. The average generalized precision of the target queries for participants not at all familiar with the topic was 17.7%, 34.5% for a little familiar and 57.1% for very familiar. Thus, those who knew the topic better got better results with the target query. The average generalized precision of the UTACLIR queries was 12.3%, 14.3% and 43.0%, respectively. These results are not very reliable, however, because the group very familiar included only five tasks out of 48.

The topic familiarity and the vocabulary familiarity had a significant effect on the performance differences in the Finnish-French test. The average performance of the target queries for participants not at all familiar with the topic was 17.7%, 34.5% for a little familiar and 57.1% for very familiar. Thus, those who knew the topic better got better results with the target query. The average generalized precision of the UTACLIR queries was 12.3%, 14.3% and 43.0%, respectively. Those very familiar with the vocabulary got better results with the target query (59.3%) than those who were a little familiar (31.9%), and the participants who were not at all familiar got the worst results (22.5%). The average generalized precision of the UTACLIR queries was 33.8% for those very familiar with the vocabulary, 19.6% for a little familiar and 2.1% for not at all familiar. The UTACLIR queries in the last group performed poorly, because the source queries formulated by the participants of this group included a lot of words missing from the translation dictionary. On the other hand, the participants who were very familiar with the vocabulary formulated source queries with words present in the dictionary, which explains their good UTACLIR results.

We found that the performance of the translated queries depended on the language pair. In the Finnish-Swedish test, the target queries performed almost equally with the UTACLIR queries. In the English-German test the UTACLIR

queries attained the best results, while in Finnish-French the target queries performed twice as well as the UTACLIR queries. The quality of Finnish-French translation is poor in GlobalDix, which explains the result. Thus, we noticed that the quality of the translation dictionary is very important for query translation: a defective dictionary does not help even those with not so good language skills.

In conclusion, query translation with good translation resources performed much better in our user test than it has performed in laboratory tests. The settings of the laboratory tests do not correspond to the real situation, where users formulate queries themselves. Both the source language and the target language queries used in laboratory tests are complete, because they are based on the topics formulated by native language speakers. On the other hand, the target queries phrased by test persons (and in a real CLIR situation) are often defective, because they are formulated by users who are not native language speakers.

In CLEF there has been an interactive track iCLEF[1] since 2001. CLIR user tests were also performed in connection with the CLARITY project (Cross Language Information Retrieval and Organisation of Text and Audio Documents). (Petrelli & al. 2006.) There are no user tests about the performance of query translation compared with the performance of direct target language retrieval, however. In addition, most of the CLIR user tests have been performed in a test collection, while Study V utilized the Web as a test bed. Thus, Study V brings valuable information about benefits of CLIR for users differing in their language skills.

[1] http://nlp.uned.es/iCLEF/index.htm

6 Discussion and conclusions

The topics of this thesis are linguistic and approximate string matching methods for controlling morphological variation in IR and CLIR. All the research questions are connected with NLP or approximate string matching in one way or the other: the studies are concerned with word normalization, translation, n-gramming, s-gramming and word form generation. Some of these subjects are studied in connection to monolingual retrieval, the others to bilingual retrieval, some to multilingual retrieval, and some join all of them.

The main contributions of this study may be summed up in five areas:

- The effect of compounds in retrieval: If compounds included in queries are decompounded in monolingual Finnish retrieval, the proximity operator is better for enveloping the parts than the synonym operator. In monolingual (Finnish, German, Swedish) retrieval, decompounding in indexing is not vital. The best results can be achieved in the lemmatized decompounded index, but the differences with other index types (lemmatized compound index, stemmed index) are mostly not significant. On the other hand, compound handling in indexing is vital in bilingual retrieval, when the source language is phrase oriented while compounds are used in the target language. All in all: decompounding in indexing is beneficial both for mono- and bilingual retrieval.
- The significance of the quality of the translation dictionary in bilingual retrieval: When two translation dictionaries are compared in laboratory tests, the better and more extensive dictionary gives remarkably better results than the poorer. When the performance of user formulated translated queries is compared with the performance of user formulated target language queries, the quality of the translation dictionary has a remarkable effect. A defective dictionary does not help even those with poor target language skills.
- Benefits of query translation for users: Query translation performed much better in our user tests than it has performed in laboratory tests. The benefits depended on language skills: users with moderate language skills benefit remarkably from query translation, while the benefits are smaller for those with good skills.
- Bilingual retrieval in an inflected index: Any kind of processing (n-gramming, s-gramming, FCG or their combination) improves the retrieval result compared with queries formulated directly from raw translations. The performance of various approaches depends on the

language pair, and thus, on the characteristics of each language. When the source language is morphologically rich (like Finnish), and the target language is quite simple (for example Swedish), s-gramming seems to perform better than FCG. In the opposite case, with a morphologically simple source language (for example English) and a morphologically rich target language (like Finnish), FCG outperforms s-gramming.

- The effect of normalization and the result list merging approach on multilingual IR: Different normalization approaches in indexing / retrieval do not have any remarkable impact on the multilingual retrieval result, even if lemmatization seems to perform slightly better than stemming. Different result list merging approaches have only minor impact on the result. This is in line with earlier research results.

The importance of NLP for information retrieval has grown when other languages than English have been brought to tests. Especially CLEF has promoted test datasets and tracks for various languages, and thus created basis for many interesting approaches and research results.

Finnish has naturally been a focus of the Finnish IR research at the University of Tampere, as well as English. These two quite different languages have clearly shown the impact of morphology on the retrieval result. Within CLEF, we have made studies in other European languages as well. The most important characteristics among languages have been found to be the morphological complexity, as well as the phrase / compound aspect. Finnish and English are opposites in these senses: English is morphologically quite a simple phrase oriented language, while Finnish is morphologically complex and compound oriented. Thus, these two languages offer a good test bed for various NLP studies.

Study III showed that compounds have only a slight impact on monolingual retrieval. We found two other reasons for performance differences between various normalization methods in monolingual retrieval: the quality of the stemmer has an impact on the result of the stemmed retrieval, and possible OOV words (words not recognized by the lemmatizer) have an impact on the result in lemmatized retrieval. The results of the bilingual runs affirmed our earlier findings about the significant effect of the phrase / compound issue on bilingual retrieval: the topics including phrases tend to fail when retrieval is performed in the index with no decompounding. This holds for all three language pairs used in the study, English-Finnish, English-German and English-Swedish. As in monolingual retrieval, the quality of the stemmer has an impact on the stemmed retrieval result.

In Study III, the source language was a phrase oriented language. What about a compound oriented language as a source language? In that case, untranslatable compounds would be decompounded and parts would be translated separately: the situation is similar to the situation of Study III. Thus, further research about phrases / compounds in bilingual retrieval may not be needed.

In Study V, query translation compared to monolingual queries formulated by test participants performed quite well. In laboratory tests, monolingual queries usually clearly outperform translated queries. The reason lies in the quality of the monolingual queries: in the laboratory tests, perfect monolingual queries are utilized, while users formulate incomplete queries. The poorer the target language skills are, the more incomplete the queries are. The next step towards user-oriented direction in CLIR research could be user tests in multilingual retrieval: there is no evidence of the benefits of this for real users. The first challenge would be to find participants for the tests: they should have some skills in several target languages.

Studies I and V showed that in bilingual retrieval, the quality of the translation dictionary has a remarkable impact on the result of translated queries. It is obvious that results are better with a translation dictionary of good quality than with a poorer one, but the extent of the impact was larger than expected. In addition, study V showed that a defective translation dictionary does not help even users with poor target language skills. In future, it would be interesting to study which actually are the differences between a good and a poor translation dictionary: is the number of translation variants the only factor, or are there differences in the quality of translation as well.

Study IV showed that there are two quite simple methods to improve the performance of bilingual retrieval in an inflected index: approximate string matching and word form generation. Both approaches performed quite well when compared to the baseline, retrieval with raw translations. However, the best inflected retrieval result was competitive with the gold standard, lemmatization, only in one language pair, English-Finnish. FCG has not yet been tested on the Web: that is one possible issue to study in future. User tests could be connected with this study.

Study II and earlier research on multilingual retrieval with result list merging showed that there is not much to do: results achieved utilizing various result list merging approaches differ only slightly from each other. On the other hand, the trend is towards multilingual indexes. Retrieval in a multilingual index is not simple, however, because formulating a multilingual query is difficult. Often retrieval has to be done separately for each target language. Thus, a multilingual retrieval task in a merged index proves to be a result list merging task. On the basis of earlier research, a simple result list merging approach, for example the round robin approach, can be adopted. Thus, interest may be directed towards other issues than result list merging, for example the one mentioned above: user tests in multilingual retrieval.

References

Abdul-Jaleel, N., Allan, J., Croft, W.B., Diaz, F., Larkey, L., Li, X., Metzler, D., Strohman, T., Turtle, H. & Wade, C. (2004). UMass at TREC 2004: Notebook. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook.*

Akmajian, A., Demers, R.A. & Harnish, R.M. (1995). *Linguistics: An introduction to language and communication*. 2st ed. MA: the MIT press.

Alkula, R. (2000). *Merkkijonoista suomen kielen sanoiksi*. Ph.D. Thesis, University of Tampere, Department of Information Studies, Acta Universitatis Tamperensis 1261.

Allan, J., Callan, J., Croft, W.B., Ballesteros, L., Byrd, D., Swan, R. & Xu, J. (1997). INQUERY does battle with TREC-6. In *Proceedings of the Sixth Text REtrieval Conference (TREC-6)*, 169-206.

Barry, C.L. (1994). User-defined relevance criteria: an exploratory study. *Journal of the American Society for Information Science* 45(3), 149–159.

Baeza-Yates, R. & Ribeiro-Neto, B. (1999). *Modern information retrieval*. England: ACM Press.

Ballesteros, L., & Croft, W. B. (1997). Phrasal translation and query expansion techniques for cross-language information retrieval. In *Proceedings of the 20th ACM SIGIR Conference on Research and Development in Information Retrieval*. New York: ACM Press, 84–91.

Belkin, N. J. (1996). Intelligent information retrieval: Whose intelligence? In *ISI '96: Proceedings of the Fifth International Symposium for Information Science*. Konstanz: Universtaetsverlag Konstanz, 25-31.

Borlund, P. (2000). *Evaluation of interactive information retrieval systems*. Åbo: Åbo Akademi University Press.

Braschler, M., Göhring, A. & Schäuble, P. (2003). Eurospider at CLEF 2002. In C. Peters, M. Braschler, J., Gonzalo & M. Kluck (Eds.), *Advances in Cross-Language Information Retrieval.* Lectures in computer science 2785. Germany: Springer-Verlag, 164-174.

Braschler M. & Ripplinger B. (2004). How effective is stemming and decompounding for German text retrieval? *Information Retrieval* 7(3-4), 291-316.

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* 30(1-7), 107-117.

Broglio, J., Callan, P. & Croft, W.B. (1994). INQUERY system overview. In *Proceedings of the TIPSTER Text Program (Phase I)*, 47-67.

Callan, J.P., Croft, W.B. & Harding, S.M. (1992). The INQUERY retrieval system. In *Proceeding of the Third International Conference on Database and Expert Systems Applications*. Springer-Verlag, 78-83.

Callan, J.P., Lu, Z. & Croft, W.B. (1995). Searching distributed collections with inference networks. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retriev*al. Seattle, WA: ACM Press, 21-28.

Carterette, B. & Jones, R. (2008). Evaluating search engines by modeling the relationship between relevance and clicks. In J.C. Platt, D. Koller, Y. Singer & S.

Roweis (Eds.), *Advances in Neural Information Processing Systems* 20. MIT Press, 217-224.

Chen, A. (2003). Cross-language retrieval experiments at CLEF 2002. In C. Peters, M. Braschler, J. Gonzalo & M. Kluck (Eds.), *Advances in Cross-Language Information Retrieval*. Lectures in computer science 2785. Germany: Springer-Verlag, 28-48.

Chen A. & Gey, F. (2004). Multilingual information retrieval using machine translation, relevance feedback and decompounding. *Information Retrieval* 7(1), 149-182.

Cosijn, E. & Ingwerser, P. (2000). Dimensions of relevance. *Information Processing and Management* 36, 533-550.

Dumais, S., Landauer, T. & Littman, M. (1996). Automatic cross-linguistic information retrieval using latent semantic indexin. In *SIGIR'96 Workshop on Cross-Linguistic Information Retrieval*, 16-23.

Ekmekçioglu, F.Ç., Lynch, M.F. & Willett, P. (1996). Stemming and n-gram matching for term conflation in Turkish texts. *Information Research* 1(1), http://informationr.net/ir/2-2/paper13.html

Fujii, A. & Ishikawa, T. (2000). Applying Machine Translation to Two-Stage Cross-Language Information Retrieval. In *Proceedings of the 4th Conference of the Association for Machine Translation in the Americas* (AMTA-2000), 13-24.

Fung, P. & McKeown, K. (1997). A technical word- and term-translation aid using noisy parallel corpora across language groups. *Machine Translation* 12, 53-87.

Gachot, D.A., Lange, E. & Jin, Y. (2000). The SYSTRAN NLP browser: an application of machine translation technology in cross-language information retrieval. In G. Grefenstette (Ed.), C*ross-language information retrieval*. Boston: Kluwer Academic Publishers, 105-118.

Goldsmith, J. (2001). Unsupervised learning of the morphology of a natural language. *Computational Linguistics* 27(2), 153–198.

Google. (2009). *Corporate information – technology overview*. http://www.google.com/corporate/tech.html

Harter, S.P. (1992). Psychological relevance and information science. *Journal of the American Society for Information Science* 43(9), 602-615.

Hedlund, T., Keskustalo, H., Pirkola, A., Sepponen, M. & Järvelin, K. (2001a). Bilingual tests with Swedish, Finnish and German queries: Dealing with morphology, compound words and query structuring. In C. Peters (Ed.), *Cross-Language Information Retrieval and Evaluation*. Proceedings of the CLEF 2000 Workshop. Lecture Notes in Computer Science 2069. Heidelberg: Springer, 211-225.

Hedlund, T., Keskustalo, H., Pirkola, A., Airio, E. & Järvelin, K. (2001b). UTACLIR @ CLEF 2001. In *Working Notes for CLEF 2001 Workshop*. Sophia Antipolis: European Research Consortium for Informatics and Mathematics. Available at http://www.ercim.org/publication/ws-proceedings/CLEF2/hedlund.pdf

Hedlund, T. (2002). Compounds in dictionary-based cross-language information retrieval. *Information Research* 7(2). http://InformationR.net/ir/7-2/paper128.html

Hedlund, T.,Keskustalo, H., Airio, E. & Pirkola, A. (2002). UTACLIR – an extendable query translation system. Paper presented at the *ACM SIGIR Workshop for Cross-Language Information Retrieval, August 15th 2002* in Tampere, Finland, 15-18.

Hedlund, T. (2003). *Dictionary-based cross-language information retrieval*. Ph.D. Thesis, University of Tampere, Department of Information Studies, Acta Universitatis Tamperensis 962.

Hiemstra, D., Kraaij, W., Pohlmann, R. & Westerveld, T. (2001). Translation resources, merging strategies, and relevance feedback for cross-language information retrieval. In C. Peters (Ed.), *Cross-language information retrieval and evaluation*. Lectures in computer science 2069. Germany: Springer-Verlag, 102-115.

Hollink, V., Kamps J., Monz C. & De Rijke M. (2004). Monolingual document retrieval for European languages. *Information Retrieval* 7(1): 33-52.

Huang, R., Sun, L., Li, J., Pan, L. & Zhang, J. (2007). ISCAS in CLIR at NTCIR-6: experiments with MT and PRF. In *Proceedings of NTCIR-6 Workshop*, 2007.

Hull, D. (1993). Using statistical testing in the evaluation of retrieval experiments. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*. New York: ACM, 329-338.

Ingwersen, P. (1996). Cognitive perspectives of information retrieval. *Journal of Documentation* 52(1), 3-50.

Ingwersen P. & Järvelin, K. (2005). *The turn. Integration of information seeking and retrieval in context*. Netherlands: Springer.

Jones, G. J. F. & Lam-Adesina, A.M. (2002). Combination methods for improving the reliability of machine translation based cross-language information retrieval. In *Artificial Intelligence and Cognitive Science*. Lecture Notes in Computer Science 2464. Germany: Springer-Verlag, 97-125.

Järvelin, A., Kumpulainen, S., Pirkola, A. & Sormunen, E. (2006). Dictionary-independent translation in CLIR between closely related languages. In F.M.G. de Jong & W. Kraaij (Eds.*), 6th Dutch-Belgian Information Retrieval Workshop (DIR 2006)*. Neslia Paniculata: Enschede, http://hmi.ewi.utwente.nl/dir2006/abstracts/jarvelin_paper.pdf

Järvelin, A. & Järvelin, A. (2008). Comparison of s-gram proximity measures in out-of-vocabulary word translation. To appear in the *Proceedings of the 15th International Symposium on String Processing and Information Retrieval*.

Järvelin, K. & Kekäläinen, J. (2000). IR evaluation methods for retrieving highly relevant documents. In N. Belkin, P. Ingwersen & M-K. Leong (Eds.), *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY: ACM Press, 41-48.

Järvelin, K. & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems* 20(4), 442-446.

Karlsson, F. (1985). Preface. In F. Karlssond (Ed.), *Computational Morphosyntax. Report on Research 1981-84*. Publications No. 13 1985. Finland: University of Helsinki, Department of General Linguistics, iii-viii.

Karlsson, F. (1994). *Yleinen kielitiede*. Finland: Yliopistopaino.

Kekäläinen, J. & Järvelin, K. (1998). The impact of query structure and query expansion on retrieval performance. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson & J. Zobel (Eds.), *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York: ACM, 130-137.

Kekäläinen, J. & Järvelin, K. (2002a). Using Graded Relevance Assessments in IR Evaluation. *Journal of the American Society for Information Science and Technology* 53(13), 1120-1129.

Kekäläinen, J. & Järvelin, K. (2002b). User-oriented evaluation methods for information retrieval: A case study based on conceptual models for query expansion. In G. Lakemeyer, & B. Nebel (Eds.), *Exploring artificial intelligence in the new millennium*. San Francisco: Morgan Kaufman Publishers, 355-379.

Kettunen, K. (2004). Covering the morphological variation of Finnish query nouns in a probabilistic best-match system. In *Proceeding of The First Baltic Conference, Human Language technologies - The Baltic Perspective*. Riga, Latvia. April 21-22, 2004, 73 - 80.

Kettunen, K. (2006). Developing an automatic linguistic truncation operator for best-match retrieval of Finnish in inflected word form text database indexes. *Journal of Information Science* 32(5), 465-479

Kettunen K. & Airio E. (2006). Is a morphologically complex language really that complex in full-text retrieval? In T. Salakoski, F. Ginter, S. Pyysalo & T. Pahikkala (Eds.), *Advances in Natural Language Processing*, LNAI 4139. Berlin Heidelberg: Springer-Verlag, 411-422.

Kettunen, K. (2007). *Reductive and generative approaches to morphological variation of keywords in monolingual information retrieval*. Ph.D. Thesis, University of Tampere, Department of Information Studies, Acta Universitatis Tamperensis 1261.

Kettunen, K., Airio, E. & Järvelin, K. (2007). Restricted inflectional form generation in management of morphological keyword variation. *Information Retrieval* 10(4-5), 415-444.

Kettunen, K. (2008). Automatic generation of frequent case forms of query keywords in text retrieval. In Nordström, B. and Ranta, A. (Eds.), *Advances in Natural Language Processing*, LNAI 5221, 222-236 (to appear).

Kilgarriff, A. (1992). *Polysemy*. Ph.D. Thesis, University of Sussex, UK.

Kishida, K. (2005). Technical issues of cross-language information retrieval: a review. *Information Processing & Management* 41(3), 433-455.

Koskenniemi, K. (1983). *Two-level Morphology: A General Computational Model for Word-Form Recognition and Production*. Ph.D. Thesis, University of Helsinki, Department of General Linguistics.

Kraaij, W. (1996). Viewing stemming as recall enhancement. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retriev*al. Seattle, WA: ACM Press, 40-48.

Kraaij, W. (2004). *Variations on language modelling for information retrieval*. Haag: CTIT Ph. D. series No. 04-62.

Krovetz, R. (1993). Viewing morphology as an inference process. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retriev*al. Seattle, WA: ACM Press, 191-202.

Lancaster, F.W (1968). *Information retrieval systems*. New York: John Wiley & Sons, Inc.

Larkey, L.S, Ballesteros, L. & Connell M. (2002). Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis. In *Proceedings of the 25th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*. Seattle, WA: ACM Press, 275-282.

Lemur. (2008). *The Lemur Toolkit for language modelling and information retrieval*. http://www.lemurproject.org/

Lennon, M., Peirce, D.S., Tarry, B.D. & Willet, P. (1981). An evaluation of some conflation algorithms for information retrieval. *Journal of Information Science* 3, 177-183.

Martínez-Santiago, F., Ureña-López, L. & Martín-Valvidia, M. (2006). A merging strategy proposal. *Information Retrieval* 9(1), 71-93.

McNamee, P., Mayfield, J. & Piatko, C. (2000). A language-independent approach to European text retrieval. In *The Working Notes of the CLEF-2000 Workshop*, Lisbon, Portugal, September 2000.

Monz, C. & De Rijke, M. (2002). Shallow Morphological Analysis in Monolingual Information Retrieval for Dutch, German, and Italian. In C. Peters, M. Braschler, J. Gonzalo & M. Kluck (Eds.), *Evaluation of Cross-Language Information Retrieval Systems*. Lectures in computer science 2406. Germany: Springer-Verlag, 1519-1541.

Moulinier, I & Molina-Salgado, H. (2003). Thomson legal and regulatory experiments for CLEF 2002. In C. Peters, M. Braschler, J. Gonzalo & M. Kluck (Eds.), *Advances in Cross-Language Information Retrieval*. Lectures in computer science 2785. Germany: Springer-Verlag, 155-163.

Navarro, G. (2001). A guided tour to approximate string matching. *ACM Computing Surveys (CSUR)* 33(1), 31-88.

Oard, D.W. (1997). Alternative approaches for cross-language text retrieval. In *AAAI Spring Symposium on Cross-Language Text and Speech Retrieval, Electronic Working Notes*. http://www.ee.umd.edu/medlab/filter/sss/papers/oard/paper.html

Oard, D.W. (1998). A comparative study of query and document translation for cross-language information retrieval. In *Proceedings of the 3rd Conference of the Association for Machine Translation in the Americas on Machine Translation and the Information Soup.* Lecture Notes In Computer Science 1529. London: Springer, 472 – 483.

Over, P. (1997). TREC-5 Interactive Track Report. In E. Voorhees & D. Harman (Eds.), *Proceedings of the Fifth Text Retrieval Conference (TREC-5)*. NIST Special Publication 500-238, 29-56.

Palmer, D.D. & Ostendorf, M. (2005). Improving out-of-vocabulary name resolution. *Computer Speech & Language* 19(1), 107-128.

Petrelli, D., Levin, S., Beaulieu, M. & Sanderson, M. (2006). Which user interaction for cross-language information retrieval? Design issues and reflections. *Journal of the American Society for Information Science and Technology* 57(5), 709-722.

Pirkola, A. (1999). *Studies on linguistic problems and methods in text retrieval: the effects of anaphor and ellipsis resolution in proximity searching, and translation and query structuring methods in cross-language retrieval*. Ph.D. Thesis, University of Tampere, Department of Information Studies, Acta Universitatis Tamperensis 672.

Pirkola, A. (2001). Morphological typology of languages for IR. *Journal of Documentation* 57(3), 330-348.

Pirkola, A., Hedlund, T., Keskustalo, H. & Järvelin, K. (2001). Dictionary-based cross-language information retrieval: problems, methods, and findings. *Information Retrieval 4(3-4),* 209-230.

Pirkola, A., Keskustalo, H., Leppänen, E., Känsälä, A-P. & Järvelin, K. (2002). Targeted s-gram matching: a novel n-gram matching technique for cross- and monolingual word form variants. *Information Research* 7(2). http://informationr.net/ir/7-2/paper126.html

Pirkola, A., Puolamäki, D. & Järvelin, K. (2003). Applying query structuring in cross-language retrieval. *Information Processing & Management* 39(3), 391- 402.

Pirkola, A., Toivonen, J., Keskustalo, H. & Järvelin, K. (2006). FITE-TRT: A High Quality Translation Technique for OOV Words. In R.L. Wainwright & al. (Eds.), *Proceedings of the 21st Annual ACM Symposium on Applied Computing.* Dijon, France, 1043-1049.

Pohlmann, R. & Kraaij, W. (1996). Improving the precision of a text retrieval system with compound analysis. In J. Landsbergen, J. Odijk, K. van Deemter & G. Veldhuijzen van Zanten (Eds.), *Proceedings of the 7th Computational Linguistics in the Netherlands Meeting (CLIN 1996)*, 115–129.

Popovič M. & Willet P. (1992). The effectiveness of stemming for natural-language access to Slovene textual data. *Journal of the American Society for Information Science* 43(5), 384-390.

Porter, M. (1981). *Snowball: A language for stemming algorithms.* http://snowball.tartarus.org/texts/introduction.html.

Powell, A., French J., Callan J, Connell, M. & Viles, C. (2000). The impact of database selection on distributed searching. In *Proceedings of the 23^{trd} Annual International ACM SIGIR Conference on Research and Development in Information Retriev*al. New York, NY: ACM Press, 232–239.

Robertson, A.M. & Willet, P. (1998). Applications of n-grams in textual information systems. *Journal of Documentation* 54(1), 48-69.

Robins, D. (2000). Interactive information retrieval: context and basic notions. *Informing Science* 3(2), 57-61.

Rosemblat, G., Gemoets, D., Browne, A.C. & Tse, T. (2003). Machine translation-supported cross-language information retrieval for a consumer health resource. In *Proceedings of AMIA Symposium 2003*, 564–8.

Saracevic, T. (1970). The notion of "relevance" in information science. In T. Saracevic (Ed.), *Introduction to information science*. New York: R.R.Bowker Company, 111-154.

Saracevic, T. (1996). Relevance reconsidered. In *Information science: Integration in perspectives. Proceedings of the Second Conference on Conceptions of Library and Information Science (CoLIS 2)*. Copenhagen, Denmark, 201-218.

Saracevic, T. (1997). The stratified model of information retrieval interaction: Extension and application. In *Proceedings of the 60th Annual Meeting of the American Society for Information Science 34*, 313-327.

Saracevic, T. (2007). Relevance: A review of the literature and a framework for thinking on the notion in information science. Part II: nature and manifestations of relevance. *Journal of the American Society for Information Science and Technology* 58(3), 1915-1933.

Schamber, L. (1994). Relevance and information behavior. *Annual Review of Information Science and Technology* 29, 3-48.

Schamber, L., Eisenberg, M.B. & Nilan, M.S. (1990). A re-examination of relevance: Toward a dynamic, situational definition. *Information Processing & Management* 26(6), 755-776

Sheridan, P. & Smeaton, A.F. (1992). The application of morpho-syntactic language processing to effective phrase matching. *Information Processing and Management* 28(9), 349-369.

Smucker M., Allan, J. & Carterette. B. (2007). A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. Lisbon, Portugal, 623-632.

Sormunen, E. (1994). *Vapaatekstihaun tehokkuus ja siihen vaikuttavat tekijät sanomalehtiaineistoa sisältävässä tekstikannassa*. Licentiate thesis at University of Tampere.

Sormunen, E. (2002). Liberal relevance criteria of TREC - counting on negligible documents? In M. Beaulieu & al. (Eds), *Proceedings of the 25th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*. August 11-15, 2002, Tampere, Finland. Special Issue of SIGIR Forum 36, 324-330.

Spink, A. (1997). Study of interactive feedback during mediated information retrieval. *Journal of the American Society for Information Science 48*(5), 382-394.

Talvensaari, T., Laurikkala, J., Järvelin, K. & Juhola M. (2006). A study on automatic creation of a comparable document collection in cross-language information retrieval. *Journal of Documentation* 62(3), 372-387.

Talvensaari, T., Laurikkala, J., Järvelin, K. & Juhola M. (2007). Creating and exploiting a comparable corpus in cross-language information retrieval. *ACM Transactions on Information Systems (ACM TOIS)* 25 (1), article 4.

Talvensaari, T. (2008a). *Comparable corpora in cross-language information retrieval*. Ph.D. Thesis, University of Tampere, Department of computer sciences, A-2008-7.

Talvensaari, T. (2008b). Effects of aligned corpus quality and size in corpus-based CLIR. In I. Ruthven & al. (Eds.), *Proceedings of the 30th European Conference on*

Information Retrieval (ECIR 2008). Lecture Notes in Computer Science 4956. Heidelberg: Springer, 114-125.

Talvensaari, T., Pirkola, A., Järvelin, K., Juhola, M. & Laurikkala, J. (2008). Focused Web crawling in the acquisition of comparable corpora. *Information Retrieval* 11(5), 427-445.

Tang. T.T, Craswell, N., Hawking, D., Griffiths, K. & Cristensen, H. (2006). Quality and relevance of domain-specific search: A case study in mental health. *Information Retrieval* 9(2), 207-225.

Voorhees, E. Evaluation by highly relevant documents. (2001). In W.B. Croft & al. (Eds.), *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* New York: ACM Press, 74-82.

Wilson, P.W. (1973). Situational relevance. *Information Storage and Retrieval* 9, 457-471.

Wu, S. & Crestani, F. (2008). Ranking Retrieval Systems with Partial Relevance Judgements. *Journal of Universal Computer Science* 14(7), 1020-1030.

Yamabana, K., Muraki, K., Doi, S. & Kamei, S. (2000). A language conversion front-end for cross-language information retrieval. In G. Grefenstette (Ed.), *Cross-language information retrieval*. Boston: Kluwer Academic Publishers, 93-104.

Yang C.C. & Kar Li, K.W. (2004). Building parallel corpora by automatic title alignment using length-based and text-based approaches**.** *Information Processing & Management* 40(6), 939-955.

Zobel, J. & Dart, P. (1995). Finding approximate matches in large lexicons. *Software – practice and experience* 25(3), 331-345.

Study I

Eija Airio, Heikki Keskustalo, Turid Hedlund & Ari Pirkola. 2003. UTACLIR @ CLEF 2002 – Bilingual and multilingual runs with a unified process. In Peters, Kluck, Gonzalo: Advances in cross-language information retrieval. Results of the Cross-Language Evaluation Forum – CLEF 2002. Lecture Notes in Computer Science 2785, Springer Verlag , 91-100.

UTACLIR @ CLEF 2002 - Bilingual and Multilingual Runs with a Unified Process

Eija Airio, Heikki Keskustalo, Turid Hedlund, and Ari Pirkola

Department of Information Studies
University of Tampere, Finland

{eija.airio,heikki.keskustalo}@uta.fi
turid.hedlund@shh.fi
pirkola@tukki.jyu.fi

Abstract. The UTACLIR system of University of Tampere uses a dictionary-based CLIR approach. The idea of UTACLIR is to recognize distinct source key types and process them accordingly. The linguistic resources utilized by the framework include morphological analysis or stemming in indexing, normalization of topic words, stop word removal, splitting of compounds, translation utilizing bilingual dictionaries, handling of non-translated words, phrase composition of compounds in the target language, and constructing structured queries. UTACLIR was shown to perform consistently with different language pairs. The greatest differences in performance are due to the translation dictionary used.

1 Introduction

The cross-language information retrieval (CLIR) task in CLEF is bilingual or multilingual. In the first task, searching is on a target collection of documents written in the same language, while the other searches a collection of documents in written in multiple languages.

One of the main approaches used in CLIR is the dictionary-based strategy, which means utilizing bilingual dictionaries in translation. University of Tampere has developed a translation and query formulation tool, UTACLIR, which is based on the use of bilingual dictionaries and other linguistic resources.

The following kinds of difficulties can occur in dictionary-based CLIR. First, the translation may be problematic. Entries in dictionaries are typically in a normalized form, therefore the source words should also be normalized. However, morphological analyzers are not available for all languages, or they are expensive. Second, poor coverage of a dictionary may cause problems. If important topic words remain untranslated, the retrieval performance will be poor. Third, languages containing compounds (multiword expressions in which component words are written together) are difficult to handle in CLIR. Dictionaries never contain all possible compounds in compound-rich languages. Therefore, morphological decomposition of compounds into constituents and their proper translation is important [1].

Retrieval topics often contain proper names in inflected form. In some cases even the base form of a proper name varies over languages because of differences in transliteration. Also technical terms are often absent in dictionaries. However, as their appearance tends to be quite similar in different languages, approximate string matching techniques, like n-gram based matching, can be applied when handling them [1].

Translation ambiguity is one of the main problems in dictionary-based CLIR. Dictionaries may contain many possible translation variants for a given source word. This can introduce noise in the query. Since queries tend to have a natural disambiguation effect because of other relevant contextual words, translation variants can be handled and the most relevant documents can still appear first in the result list.

In multilingual information retrieval an additional problem, the merging problem, is encountered. There are two main approaches towards solving this problem: merging result lists and merging indexes. In the first one separate indexes are built for each target language, and retrieval is performed separately from each index. The result lists must then be merged somehow. In the second approach, a common

index is built for documents representing different languages. Queries in different languages are merged to one query, and retrieval is performed from the merged index.

2 The UTACLIR Approach

The University of Tampere has developed the UTACLIR translation and query formulation system for cross-language information retrieval. We participated in CLEF 2000 and 2001 utilizing the UTACLIR process, which consisted of separate, but similar programs for three language pairs: Finnish - English, Swedish - English and German - English. In CLEF 2002 we used a new version of UTACLIR: the same program used external language resources for all the different language pairs.

The source word processing of UTACLIR can be described on a general level as follows (see Figure 1). First the topic words are normalized with a morphological analyser, if possible, and after that source stop words are removed. Then translation is attempted. If a translation or translation variants are found, the further handling of the translated words depends on the form of the index. The target query words must be stemmed if the index is stemmed, and accordingly they must be normalized with a morphological analyzer if the index is morphologically normalized. Since stop word lists are in a morphologically normalized form, stop word removal can only be done when using the same method in target language queries.

The untranslatable words are mostly compound words, proper names and technical terms. In many cases these words are spelling variants of each other in different languages, thus allowing the use of approximate string matching techniques. The techniques utilized by the UTACLIR process are language-independent [2]. The best matching strings can be selected from the target index, enveloped with a synonym operator and added to the query.

Structuring of queries using the synonym operator, which means grouping of the target words derived from the same source word into the same facet, is applied in the UTACLIR system. This has proved to be an effective strategy in CLIR in earlier studies [3].

Finally, UTACLIR has a special procedure for untranslatable compounds. They are first split into their constituents and then the components are translated separately. Translated parts are enveloped with a proximity operator [4].

It is possible to use parallel resources in the UTACLIR system. In that case the input codes denote not only the source and target language, but also specify the resource used. For example, if we have three different English - Finnish bilingual dictionaries in use, we can easily compare their performance as components of UTACLIR.

3 Runs and Results

The University of Tampere participated in CLEF 2002 in the Finnish monolingual task, the English - Finnish, English - French and English - Dutch tasks, and the multilingual task.

In this section, we first describe the language resources used, then the collections, and then the indexing strategy adopted. Finally, we report results of our official runs plus additional monolingual, bilingual and multilingual runs.

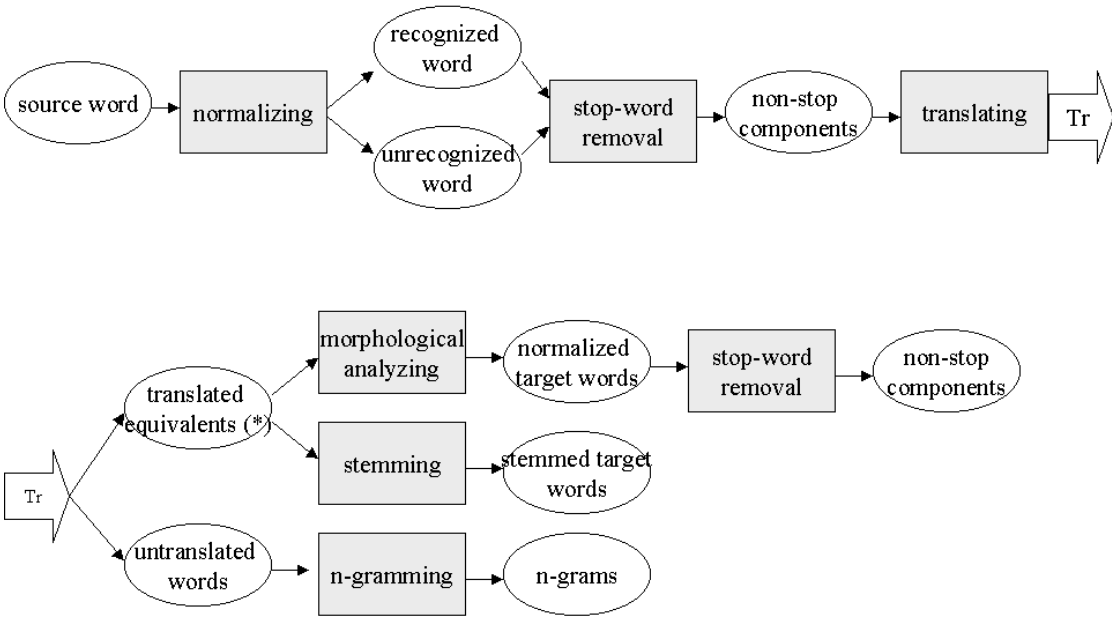

Fig. 1. An overview of processing a word in the UTACLIR process with English as source language. (Depending on the target language, either morphological analysis or stemming is performed.)

3.1 Language Resources

- Motcom GlobalDix multilingual translation dictionary (18 languages, total number of words 665 000) by Kielikone plc. Finland
- Motcom English . Finnish bilingual translation dictionary (110 000 entries) by Kielikone plc. Finland
- Morphological analysers FINTWOL, GERTWOL and ENGTWOL by Lingsoft plc. Finland
- Stemmers for Spanish and French, by ZPrise
- A stemmer for Italian, by the University of Neuchatel
- A stemmer for Dutch, by University of Utrecht
- An English stop word list, created on the basis of InQuery's default stop list for English
- A Finnish stop word list, created on the basis of the English stop list
- A German stop word list, created on the basis of the English stop list

3.2 Test Collections

The following test collections were used in the CLEF 2002 runs: English *LA Times*, Finnish *Aamulehti*, French *Le Monde*, French *SDA*, German *Der Spiegel*, German *SDA*, Italian *La Stampa*, Italian *SDA*, Spanish *EFE*, Dutch *NRC Handelsblad* and Dutch *Algemeen Dagblad*. We had to exclude German *Frankfurter Rundschau* from the official runs because of indexing problems, but we have made additional runs later in which it is present. Next, the indexing of the databases is described.

Lingsoft's morphological analyser FINTWOL was utilized in indexing the Finnish dataset, and GERTWOL in indexing the German datasets. As we did not have morphological analysers for Spanish, Italian, French and Dutch, we indexed those databases by utilizing stemmers. We used Zprise's Spanish stemmer, Zprise's French stemmer, the Italian stemmer granted by the University of Neuchâtel and the Dutch stemmer granted by University of Utrecht.

The *InQuery* system, provided by the Center for Intelligent Information Retrieval at the University of Massachusetts, was utilized in indexing and building the databases, as well as a retrieval system.

3.3 Monolingual Runs

We participated in CLEF 2002 with two monolingual runs, both in Finnish. The approach used in these runs was similar to our bilingual runs, only excluding translation. In the first run topic words were normalized using Lingsoft's morphological analyser FINTWOL. Compounds were split into their constituents. If a word was recognized by FINTWOL, it was checked against the stop word list, and the result (the normalized word, or nothing in the case of stop word) was processed further. If the word was not recognized, it was n-grammed. The n-gram function compares the word with the database index contents. The system utilized returns one best match form among morphologically recognized index words, and one best match form among non-recognized index words, and combines them with InQuery's synonym operator (#syn operator, see [5]).

Table 1. Average precision for Finnish monolingual runs using synonym and uw3 operator

	Average precision (%)	Difference (% units)	Change (%)
N-gramming and Synonym operator	27.0		
N-gramming and uw3 operator	35.2	+8.2	+30.4
No N-gramming, Synonym operator	24.0		
No N-gramming, uw3 operator	32.0	+8.0	+33.3

The second monolingual Finnish run was similar to the first one, except no n-gramming was done. Unrecognised words were added to the query as such. There was no big difference in performance between the results of these two Finnish monolingual runs.

Finnish is a language rich in compounds. Parts of a compound are often content bearing words [6]. Therefore, in a monolingual run it is reasonable to split a compound into its components, normalize the components separately, and envelope the normalized components with an appropriate operator. In the original run, we used the synonym operator in the monolingual runs for this purpose instead of the proximity operator, which turned out to be an inferior approach.

We made an additional run in order to gain a more precise view of the effect of the synonym operator in the compounds compared with the proximity operator. We replaced the synonym operator with InQuery's #uw3 operator (proximity with the window size 3) in order to connect the compound components. We compared these new results to the corresponding results in our CLEF runs (see Table 1). Average precision of this additional run was 30.4 % better in the run using n-grams, and 33.3 % better in the run using no n-grams. We can conclude that requiring all the parts of the compound to be present in the query is essential to get better results.

We made monolingual English, German, Spanish, Dutch, French and Italian runs as baseline runs for the bilingual runs. These monolingual runs are also reported in the next section.

3.4 Bilingual Runs

We participated in CLEF 2002 with three bilingual runs: English - Finnish, English - Dutch and English - French. The English - Dutch run was not reported in the CLEF Working Notes because of a severe failure in the indexing of the Dutch database. In this paper we will report the results of an additional run we made later. Bilingual English - German, English - Italian and English - Spanish runs were also done for CLEF 2002 multilingual task. We also made additional English - German and English - Finnish runs. In the first one the Frankfurter Rundschau dataset, which was not available

during the CLEF runs, was added to the index. The average precision of the official run was 13.5 %, and that of the additional run 21.2 % In the additional English - Finnish run untranslatable words were added to the query in two forms: as such and preceded by the character '@' (unrecognised words are preceded by '@' In the index). The average precision of the official run was 20.2 %, and that of the additional run 24.6%.

Next we will report the performance of English - Finnish, English - German, English - Spanish, English - Dutch, English - French and English - Italian runs in order to clarify the performance of UTACLIR. Monolingual Finnish, German, Spanish, Dutch, French and Italian runs were made to provide the baseline runs.

The topics were in English in all cases, so the beginning of the process was similar in every language: topic words were normalized using ENGTWOL and after that the source stop words were removed. The GlobalDix dictionary was then used to translate the normalized source words into the target languages. As we have morphological analysers for Finnish and German (FINTWOL and GERTWOL by Lingsoft), they were used to normalize the translated target words. However, for Spanish, French, Italian and Dutch we did not have morphological analysers, thus we utilized stemmers instead. We used ZPrise's Spanish and French stemmers, the Italian stemmer of the University of Neuchâtel and a Dutch stemmer of the University of Utrecht. Target stop word removal was done only for morphologically analysed target queries (Finnish and German runs), as we did not have stop lists for stemmed word forms.

The average precision of the bilingual runs varies between 20.1 % (English - Italian run) and 24.6 % (English - Finnish run) (Table 2). The performance of UTACLIR is quite similar with all the six language pairs, but there are big differences in the monolingual runs between the languages. The average precision of the monolingual runs varied from 24.5 % (French) to 38.3 % (Spanish). The differences between the baseline run and the bilingual run vary from 0 % (English - French) to -42.4 % (English - Italian).

We had a beta-version of UTACLIR in use during the runs. There were some deficiencies compared to the old version, because all the features of UTACLIR were not yet implemented in the new one. Splitting of compounds was not yet implemented, and non-translated words were handled by the n-gram method only in German as the target language. We also did not utilize target stop word removal in the case of stemming. Our stop word lists consist of morphologically normalized words at the moment, thus they could not be used as such to remove the stemmed forms. Implementing the Italian and Spanish dictionaries was also not complete when making the runs. Thus, we can expect better results with those languages after some development of UTACLIR.

The translation strings given by the GlobalDix sometimes contained lots of garbage. This had an impact on the result of the bilingual runs. We also have a parallel English - Finnish dictionary in use, which is a MOT dictionary with 110 000 entries (compared to 26 000 entries of GlobalDix). Both dictionaries are from the same producer, Kielikone plc. We made additional English - Finnish runs to clarify the effect of the dictionary on the result. (Table 3). The result was 32.5 % better using another translation source than in the original CLEF result. Thus, as expected, the use of a more extensive dictionary clearly improved the results.

As we did not have parallel resources for all the other languages, we could not compare the effect of the dictionary on these results.

Table 2. Average precision for monolingual and bilingual runs

	Average precision (%)	Difference (% units)	Change (%)
Monolingual Finnish	35.2		
Bilingual English.Finnish	24.6	-10.6	-30.1
Monolingual	29.9		

German			
Bilingual English-German	21.2	-8.7	-29.1
Monolingual Spanish	38.3		
Bilingual English -Spanish	21.8	-16.5	-43.1
Monolingual Dutch	32.2		
Bilingual English-Dutch	21.3	-10.9	-33.9
Monolingual French	24.5		
Bilingual English-French	23.9	-0.6	0
Monolingual Italian	34.9		
Bilingual English - Italian	20.1	-14.8	-42.2

Table 3. Average precision for English - Finnish bilingual runs using alternative resources

	Average precision (%)	Difference (% units)	Change (%)
GlobalDix	24.6		
MOT bilingual	32.6	+8.0	+32.5

3.5 Multilingual Runs

There are several possible result merging approaches. The simplest of them is *the Round Robin approach*, where a line of every result set is taken, one by one from each. This is the only possibility if only document rankings are available. This is not a very good approach, because collections seldom include the same number of relevant documents. If document scores are available, it is possible to use more advanced approaches. In *the raw score approach* the document scores are used as such as a basis of merging. The disadvantage of this approach is that the scores from different collections are not comparable. *The normalized score approach* tries to overcome this problem. Normalizing can be done for example by dividing the document score by the maximum score of the collection [7].

Table 4. Average precision for official and additional multilingual runs with different merging Strategies

	Official CLEF 2002 runs	Additional runs	Difference (% units)	Change (%)
Raw score approach	16.4	18.3	+1.9	+11.6
Round robin approach	11.7	16.1	+4.4	+37.6

In the first official run we applied the raw score merging method, and in the second run the round robin method. In the official CLEF runs the material of Frankfurter Rundschau was not available. We made additional runs where it was included. The results of the additional runs were somewhat better than official runs (see table 4).

The average precision of the bilingual runs present in the multilingual runs can be seen in Table 2. The average precision for the monolingual English run was 47.6 %.

The results of our multilingual runs are quite poor compared to the corresponding bilingual runs. We must concentrate on different result merging techniques to achieve better results.

4 Discussion and Conclusion

The problems of dictionary-based cross-lingual information retrieval include word inflection, translation ambiguity, translation of proper names and technical terms, compound splitting, using of a stop list and query structuring. Multilingual tasks have an additional problem: result merging or index merging. The UTACLIR process can handle the first four problems. The merging problem is independent of the UTACLIR process for query translation and formulation. The merging problem has to be solved, however, when we are dealing with multilingual tasks.

Our test runs showed that the translation dictionary has a significant effect on CLIR performance. The multilingual dictionary we used that included several languages was not as extensive as separate bilingual dictionaries. The result of the multilingual task using a *result merging* approach was always worse than the results of the corresponding bilingual tasks. If the results of bilingual tasks are poor, it is impossible to achieve good multilingual results by merging. If an *index merging* approach is followed, there is no additional merging phase which would ruin results. However, defective translation also causes problems in this case.

Previous research results show that better results should be achieved by applying result merging than by index merging approach [8], [9]. However, as one goal of multilingual information retrieval is also to create functional systems for Internet, the index merging approach should be further studied in the future.

Acknowledgements

The *InQuery* search engine was provided by the Center for Intelligent Information Retrieval at the University of Massachusetts.

- ENGTWOL (Morphological Transducer Lexicon Description of English): Copyright (c) 1989-1992 Atro Voutilainen and Juha Heikkilä.
- FINTWOL (Morphological Description of Finnish): Copyright (c) Kimmo Koskenniemi and Lingsoft plc. 1983-1993.
- GERTWOL (Morphological Transducer Lexicon Description of German): Copyright (c) 1997 Kimmo Koskenniemi and Lingsoft plc.
- TWOL-R (Run-time Two-Level Program): Copyright (c) Kimmo Koskenniemi and Lingsoft plc. 1983-1992.
- GlobalDix Dictionary Software was used for automatic word-by-word translations. Copyright (c) 1998 Kielikone plc, Finland.
- MOT Dictionary Software was used for automatic word-by-word translations. Copyright (c) 1998 Kielikone plc, Finland.

References

[1] Hedlund, T., Keskustalo, H., Pirkola, A., Airio, E., Järvelin K.: Utaclir @ CLEF 2001 . Effects of compound splitting and n-gram techniques. Evaluation of Cross-language Information Retrieval Systems. Lecture Notes in Computer Science; Vol. 2406. Springer-Verlag, Berlin Heidelberg New York (2002) 118-136

[2] Pirkola, A., Keskustalo, H., Leppänen, E., Känsälä, A., Järvelin, K.: Targeted s-gram matching: a novel n-gram matching technique for cross- and monolingual word form variants. Information Research, **7**(2) (2002), http://InformationR.net/ir/7-2/paper126.html

[3] Pirkola, A.: The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. Proceedings of the 21st ACM/SIGIR Conference (1998) 55-63

[4] Hedlund, T., Keskustalo, H., Pirkola, A., Airio, E., Järvelin, K.: UTACLIR @ CLEF 2001: New features for handling compound words and untranslatable proper names. Working Notes for the CLEF 2001 Workshop, Italy (2001) 118-136, http://www.ercim.org/publication/ws-proceedings/CLEF2/hedlund.pdf

[5] Kekäläinen, J., Järvelin, K.: The impact of query structure and query expansion on retrieval performance. Proceedings of the 21st ACM/SIGIR Conference (1998) 130.137
[6] Hedlund, T., Pirkola, A., Keskustalo, H., Airio, E.: Cross-language information retrieval: using multiple language pairs. Proceedings of ProLISSA. The Second Biannual DISSAnet Conference, Pretoria (2002) 24-25 October, 2002
[7] Callan, J.P., Lu, Z., Croft, W.B.: Searching distributed collections with inference networks. Proceedings of the 18tth ACM/SIGIR Conference (1995) 21-28
[8] Chen, A.: Cross-language retrieval experiments at CLEF 2002. Working Notes for the CLEF 2002 Workshop, Italy (2002) 5-20, http://clef.iei.pi.cnr.it:2002/workshop2002/WN/01.pdf

[9] Nie, J., Jin, F.: Merging different languages in a single document collection. Working Notes for the CLEF 2002 Workshop, Italy (2002) 59-62, http://clef.iei.pi.cnr.it:2002/workshop2002/WN/6.pdf

Study II

Eija Airio, Heikki Keskustalo, Turid Hedlund & Ari Pirkola. 2004. The impact of word normalization methods and merging strategies on multilingual IR. In Peters, Gonzalo, Braschler, Kluck: Comparative Evaluation of multilingual information access systems. Lecture notes in Computer Science 3237, Springer Verlag, 74-84.

The Impact of Word Normalization Methods and Merging Strategies on Multilingual IR

Eija Airio, Heikki Keskustalo, Turid Hedlund, and Ari Pirkola
Department of Information Studies,
University of Tampere, Finland

{eija.airio, heikki.keskustalo}@uta.fi,
turid.hedlund@shh.fi, pirkola@tukki.jyu.fi

Abstract. This article deals with both multilingual and bilingual IR. The source language is English, and the target languages are English, German, Finnish, Swedish, Dutch, French, Italian and Spanish. The approach of separate indexes is followed, and four different merging strategies are tested. Two of the merging methods are classical basic methods: the Raw Score method and the Round Robin method. Two simple new merging methods were created: the Dataset Size Based method and the Score Difference Based method. Two kinds of indexing methods are tested: morphological analysis and stemming. Morphologically analyzed indexes perform a slightly better than stemmed indexes. The merging method based on the dataset size performs best.

1 Introduction

Word inflection is a well known source of problems in information retrieval. In the case words are indexed in their inflected forms, the most common word forms should be added into the query or truncation should be applied. Another basic approach to solve the inflection problem is to normalize index words, and respectively, to normalize query words.

The two basic approaches to index a multilingual document collection are to build a separate index for each document language, and to build a common multilingual index. If the first approach is followed, retrieval must be performed separately from each index. Subsequently, the result lists have to be merged.

The impact of two different word normalizing approaches (with respect to individual indexes) and of four result merging strategies on the retrieval result are investigated in this article. Our approach utilizes separate indexes. The UTACLIR query translation system is applied in the tests. UTACLIR is developed at University of Tampere (UTA) [1]. It was originally designed for the CLEF 2000 and 2001 campaigns. The system has been developed from separate programs for every language pair towards a unified system for multiple language pairs.

2 Word Normalization Methods

The area of linguistics concerned with the internal structure of words is called morphology. Inflectional morphology studies word forms and grammatical relations between words, for example plural of nouns, or tempus of verbs. Derivational morphology goes beyond the syntax: it may affect word meaning as well. The impact of morphology on information retrieval is language dependent. English, for example, has quite weak morphology, and word inflection does not have a great impact on IR.
On the other hand, there are languages with strong morphology (e.g. Hungarian, Hebrew and Finnish), which may have hundreds or thousands of word form variants. The impact of word inflection on IR is considerable in these cases [2].

The two main approaches to handle inflection are: (a) to normalize index words, or (b) to leave index words inflected and let users handle the problem. The latter approach puts the responsibility for using the right search technique on the user, exemplified by the search engines of the Internet. This is understandable because of the huge amounts and large diversity of data. Users of Internet search engines are guided either to use truncation (for example Alta Vista, http://www.altavista.com/) or to supply all requested word forms (for example Google, http://www.google.com/).

There are two kinds of word normalization methods: stemming and morphological analysis. The purpose of stemming is to reduce morphological variance of words. There are several stemming techniques. The simplest stemming algorithms only remove plural endings, while more developed ones handle a variety of suffixes in several steps [2]. Stemming is a normalization approach compatible with languages with weak morphology, because their inflection rules are easy to apply in a stemming

algorithm. Stemming may not be the best normalizing method for languages with strong morphology, because it is not possible to create simple rules for them. Morphological analyzers are more sophisticated normalizing tools. Their basis is a two-level model consisting of two major components: a lexicon system and two-level rules. Those interdependent components are based on a common alphabet, and together they form the complete description of word inflection [3].

3 Merging Methods

There are many ways to merge result lists. One of the simplest is *the Round Robin method*, which bases on the idea that document scores are not comparable acrosscollections. Because one is ignorant about the distribution of relevant documents in the retrieved lists, an equal number of documents is taken from the beginning of each result list [4].

The Raw Score method is based on the assumption that document scores are comparable across collections [4]. The lists are sorted directly according to document scores. The raw score approach has turned out to be one of the best basic methods ([5], [6], [7]).

Also different methods for normalizing the scores have been developed. A typical *Normalized Score method* is to divide the score by the maximum score of the retrieval result in each collection. Some other balancing factors can be utilized as well.

Several more sophisticated approaches have been developed, but there have not been any breakthroughs.

4 The UTACLIR Process

In the UTACLIR process, the user gives the source and the target language codes and the search request as input. The system uses external resources (bilingual dictionaries, morphological analyzers, stemmers, n-gramming functions and stop lists) according to the language codes [8].

UTACLIR processes source words as follows:

1) a word is normalized utilizing a morphological analyzer (if possible)
2) source language stop words are removed
3) the normalized word is translated (if possible)
4) if the word is translatable, the resulting translations are normalized (by a morphological analyzer or a stemmer, depending on the target language code)
5) target language stop words are removed (in the case that a morphological analyzer was applied in phase 4)
6) if the word is untranslatable in phase 4, the two most highly ranked words obtained by n-gram-matching from the target index are selected as query words.

5 Runs and Results

In this section, we first describe the language resources used, then the collections, and the merging strategies adopted in our runs. Finally, we report the results of the runs we have performed.

5.1 Language Resources

We used the following language resources in the tests:

- Motcom GlobalDix multilingual translation dictionary) by Kielikone plc. Finland (18 languages, total number of words 665,000, 25,000 English - Dutch, 26,000 English - Finnish, 30,000 English – French, 29,000 English – German, 32,000 English – Italian, 35,000 English – Spanish and 36,000 English – Swedish entries)
- Morphological analyzers FINTWOL (Finnish) GERTWOL (German), SWETWOL (Swedish) and ENGTWOL (English) by Lingsoft plc. Finland
- Stemmers for Spanish and French, by ZPrise

- A stemmer for Italian, by the University of Neuchatel
- A stemmer for Dutch, by the University of Utrecht
- SNOWBALL Stemmers for English, German, Finnish and Swedish, by Dr Martin Porter
- English stop word list, created on the basis of InQuery's default stop list for English
- Finnish stop word list, created on the basis of the English stop list
- Swedish stop word list, created at the University of Tampere
- German stop word list, created on the basis of the English stop list

5.2 Test Collections and Indexes

The test collections of the "large multilingual" (Multilingual-8) track of CLEF 2003 were used for the tests.

Twelve indexes were built for the tests. For English, Finnish, German and Swedish we built two indexes: one utilizing a stemmer, and one utilizing a morphological analyzer. For Dutch, French, Italian and Spanish we built one stemmed index each.

The *InQuery* system, provided by the Center for Intelligent Information Retrieval at the University of Massachusetts, was utilized in indexing the databases and as a test retrieval system.

5.3 Merging Methods Applied

The *Raw Score* merging method was selected for the baseline run, because the raw score method is one of the best basic methods. The *Round Robin* method was included in the tests because of its simplicity. In addition we created two novel simple merging methods: the *Dataset Size Based method* and the *Score Difference Based* method.

The *Dataset Size Based* method is based on the assumption that it is likely that more relevant documents are found in a large dataset than in a small dataset. The number of document items taken from single result sets was calculated as follows: T * n / N, where T is the number of document items per topic in the single result list (in CLEF 2003 it was 1000), n is the dataset size and N is the total number of documents (the sum of documents in all the collections). 185 German, 81 French, 99 Italian, 106 English, 285 Spanish, 120 Dutch, 35 Finnish and 89 Swedish documents were selected for every topic in these test runs.

In *Score Difference Based* method every score is compared with the best score for the topic. Only documents with the difference of scores under the predefined value are taken to the final list. This is based on the assumption that documents whose scores are much lower than the score of the top document, may not be relevant.

5.4 Monolingual and Bilingual Runs

Two monolingual runs and ten bilingual runs were made for the multilingual track (see Table 1). The monolingual runs are in English, and the retrieval was performed in both a morphologically normalized and stemmed index, respectively. Two English – Finnish, English – German and English – Swedish runs were performed (also in morphologically normalized and stemmed indexes). English – Dutch, English – French, English – Italian and English – Spanish runs were performed solely with a stemmed index.

The average precision of these runs varied from 17.4 % (English – Dutch) to 46.3 % (monolingual English with the stemmed index). The morphologically normalized English – Finnish run achieved the best result (34.0 %) among the bilingual runs.

The monolingual and bilingual runs give the possibility to compare the performance of the morphological analyzer with the performance of the stemmer. The results of the two monolingual English runs do not differ prominently from each other: the run with the stemmed index performed 1.5 % better than the run with the morphologically normalized index. The results of bilingual English – Finnish, English – German and English – Swedish runs are different. The stemmed indexes give much

worse results than morphologically normalized indexes. The difference is -29.9 % in English – Swedish runs, -17.1 in English – German runs and –44.1 % in English – Finnish runs, when the results given by the stemmers are compared with the results obtained in morphologically analyzed indexes.

Table 1. Average precision (%) of monolingual and bilingual runs (source language English)

Type of run	**Index type**	**Average Precision %**	**Change (%)**
monolingual English	morph.anal. stemmed	45.6 46.3	 +1.5
English-Finnish	morph.anal. stemmed	34.0 19.0	 -44.1
English-Swedish	morph.anal. stemmed	27.1 19.0	 -29.9
English-German	morph.anal. stemmed	31.0 25.7	 -17.1
English-Dutch	stemmed	17.4	
English-French	stemmed	32.1	
English-Italian	stemmed	30.6	
English-Spanish	stemmed	28.3	

Next, individual queries of English – Finnish, English – Swedish and English – German runs are analyzed more closely. We pay attention particularly to those queries where the morphological analyzer produced much better results than the stemmer. Two query types, where the morphological analyzer was superior to the stemmer, were found.

Phrases – Compounds (Phrases Written Together). A closer analysis of individual queries of the two English – Finnish runs shows that the greatest performance differences can be detected in the queries containing phrases. Source query number 187 includes a phrase “nuclear transport”. The parts of this phrase are translated independently. In Finnish compounds are used instead of phrases. The corresponding word in Finnish is “ydinjätekuljetus”. When *stemming* is applied during indexing, compounds are not split, so we have only the compound “ydinjätekuljetus” in stemmed form in the index. No matches are found during retrieval, because the query includes only the individual parts of the phrase, not the full compound. When indexing is performed utilizing the *morphological analyzer*, compounds are split, and the full compound as well as parts in basic form are indexed. In retrieval, parts of the phrases now match parts of the compound. See Examples 1 and 2 in the Appendix.

The same phenomenon can be seen in Query 141 in English – Swedish runs. The phrase in the source query is “letter bomb”, and the translated query includes Swedish variants for those words. The stemmed index includes only the compound “brevbomb” (letter bomb) in its stemmed form. The morphologically analyzed index includes the compound as well as its parts in basic form. See Examples 3 and 4 in the Appendix.

The English Query 184 includes the phrase ”maternity leave”. In the English – German run the parts of this phrase are translated independently into the German word “Mutterschaft” and the words “Erlaubnis verlassen zurücklassen Urlaub lassen überlassen hinterlassen”, respectively. Again, the stemmed index includes only the compound “Mutterschaftsurlaub” in its stemmed form, but the morphologically analyzed index includes the parts of the compound as well. See Examples 5 and 6 in the Appendix.

Strong Morphology. When analysing the performance of individual queries of the stemmed English – Finnish and English – Swedish runs, another basic reason for bad results can be found: strong morphology, and the inability of stemmers to cope with it. The source query 183 includes the word “remains”, which is translated into Finnish as ”tähteet maalliset jäännökset”. The word “tähteet” is further stemmed into the string “täht”. The problem is in the fact that also the word “tähti” (star) has the same stem, which causes noise in retrieval. See Example 7 in the Appendix.

A similar phenomenon can be found in the English - Swedish run in Query 148. The word “layer” is translated into the Swedish word “lager”, which is further stemmed to a string “lag”. In Swedish there is a word “lag” (law), which has the same stem “lag”. See Example 8 in the Appendix.

5.5 Multilingual Runs

There are two variables in our multilingual runs: the index type and the merging approach. The index types are (a) morphologically analyzed / stemmed, where English, Finnish, German and Swedish indexes are morphologically analyzed, while Dutch, French, Italian and Spanish indexes are stemmed, and (b) solely stemmed, where all the indexes are stemmed. The merging approaches are the Raw Score method, the Dataset Size Based method, the Score Difference Based method (with difference value 0.08) and the Round Robin method. We tested two index types and four merging approaches, thus we have eight different runs.

The differences between the results of the multilingual runs are quite minor (see Table 2). The runs with morphologically analyzed / stemmed indexes seem to perform better than the runs with solely stemmed indexes. The best result, 20.2% average precision, was achieved by the run performed in the morphologically normalized / stemmed indexes, applying the dataset size based method. The raw score method performed worst among both index types. Even the simple round robin approach produced better results than the raw score method. However, all results are within a range of 1.7%.

Table 2. Average precision (%) of multilingual runs

Index type	Merging strategy	Average precision %	Difference %	Change (%)
morphologically analyzed/ temmed (baseline)	raw score	19.8		
morphologically analyzed/ stemmed	dataset size based	20.2	+0.4	+2.0
morphologically analyzed/ stemmed	score diff. per topic	19.9	+0.1	+0.5
morphologically analyzed/ stemmed	round robin	20.1	+0.3	+1.6
solely stemmed	raw score	18.5	-1.3	-6.6
solely stemmed	dataset size based	18.7	-1.1	-5.6
solely stemmed	score diff. per topic	18.7	-1.1	-5.6
solely stemmed	round robin	18.6	-1.2	-6.1

6 Discussion and Conclusion

The combined impact of different normalizing methods, stemming and morphological analysis, on the IR performance has not been investigated widely. The reason for that is presumably the fact that English is the traditional document language in IR tests. English is a language with simple morphology, which implies that stemming is an adequate word form normalization method. Our monolingual English tests with CLEF 2003 data support this: the result with the stemmed English index is a little better than the result with the morphologically normalized index. The bilingual test we made with Finnish, German and Swedish indexes show opposite results. The results with stemmed indexes in these languages are much worse than the results with the index built utilizing a morphological analyzer. This is in line with earlier research: Braschler and Ripplinger discovered that stemming improves the results in retrieving German documents, but morphological analysis with compound splitting produces the best result [9]. When high precision is demanded, stemming is not an adequate normalizing method with languages with strong morphology, especially in compound rich languages.

Two main reasons for the success of the morphological analyzers compared with stemmers were found. First, when phrases are used in the source language while the target language uses compounds instead, stemmers do not handle properly queries including phrases. When indexing is performed utilizing a stemmer, compounds are not split, and only the full compound is indexed in stemmed form. The target query includes only the parts of the phrase translated and stemmed, and no matches are found in retrieval. However, when the morphological analyzer is utilized during indexing and the compounds are split, components of compounds are also indexed, and matches are found. Second, if the target language is morphologically rich, and the stemmer is unable to handle the morphological variation, loss of precision is presumable. The problems caused by phrases vs. compounds were found in English – Finnish, English – Swedish and English – German runs, while the problems caused by rich inflection were found only in English – Finnish and English – Swedish runs.

We had two index variables in our multilingual tests: (a) morphologically analyzed (English, Finnish, German and Swedish) / stemmed (Dutch, French, Italian and Spanish) indexes and (b) stemmed indexes. Our tests showed that runs with indexes of type (a) outperform those of (b). We cannot show that morphologically analyzed indexes always perform better in multilingual (and bilingual) runs, because we are lacking Dutch, French, Italian and Spanish morphological tools. It is possible, that as for English, also in other morphologically weak languages, stemmers are more suitable normalizing tools than morphological analyzers.

On the other hand, the most used IR systems in real life are the search engines of the Internet. They use inflected indexes, which means that the users have to handle inflection. Truncation is possible with some search engines, while others guide their users to supply all the possible forms of their search words. Loss of recall is liable, but recall may not be important in WWW searching. In many cases, precision is more important, which may be good even if the user has not perfect language skills.

In most multilingual experiments, separate indexes are created for different languages, and various result merging strategies are tested. The results of experiments with the merged index are not very promising ([10], [11]). In real life, the situation of separate indexes and result merging occurs quite rarely, however. This would be a reason to direct research towards the strategies of the merged index approach.

Acknowledgements

The *InQuery* search engine was provided by the Center for Intelligent Information Retrieval at the University of Massachusetts.

ENGTWOL (Morphological Transducer Lexicon Description of English): Copyright (c) 1989-1992 Atro Voutilainen and Juha Heikkilä.

FINTWOL (Morphological Description of Finnish): Copyright (c) Kimmo Koskenniemi and Lingsoft plc. 1983-1993.

GERTWOL (Morphological Transducer Lexicon Description of German): Copyright (c) 1997 Kimmo Koskenniemi and Lingsoft plc.

TWOL-R (Run-time Two-Level Program): Copyright (c) Kimmo Koskenniemi and Lingsoft plc. 1983-1992.

GlobalDix Dictionary Software was used for automatic word-by-word translations. Copyright (c) 1998 Kielikone plc, Finland.

MOT Dictionary Software was used for automatic word-by-word translations. Copyright (c) 1998 Kielikone plc, Finland.

This work was partly financed by CLARITY (Information Society Technologies Programme, IST-2000-25310).

References

1. Hedlund, T., Keskustalo, H., Pirkola, A., Airio, E., Järvelin, K.: Utaclir @ CLEF 2001 – Effects of Compound Splitting and N-gram Techniques. Evaluation of Cross-language Information Retrieval Systems. Lecture Notes in Computer Science; Vol. 2406. Springer- Verlag, Germany (2002) 118-136
2. Krovetz, R.: Viewing Morphology as an Inference Process. Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1993) 191–202
3. Koskenniemi, K.: Two-level Morphology: A General Computational Model for Word-Form Recognition and Production. University of Helsinki, Finland. Publications No. 11 (1983)
4. Hiemstra, D., Kraaij, W., Pohlmann, R., Westerveld, T.: Translation Resources, Merging Strategies, and Relevance Feedback for Cross-Language Information Retrieval. Cross- Language Information Retrieval and Evaluation. Lectures in Computer Science, Vol. 2069. Springer-Verlag, Germany (2001) 102-115
5. Chen, A.: Cross-language Retrieval Experiments at CLEF 2002. Working Notes for the CLEF 2002 Workshop, Italy (2002) 5-20
6. Moulinier, I., Molina-Salgado, H.: Thomson Legal and Regulatory Experiments for CLEF 2002. Working Notes for the CLEF 2002 Workshop, Italy (2002) 91-96
7. Savoy, J., Rasolofo, Y.: Report on the TREC-9 Experiment: Link-Based Retrieval and Distributed Collections. Proceedings of the Ninth Text Retrieval Conference, NIST Special Publication 500-249, Department of ommerce, National Institute of Standards and Technology (2001) 579–588
8. Airio, E., Keskustalo, H., Hedlund, T., Pirkola, A. UTACLIR @ CLEF2002 – Bilingual and Multilingual Runs with a Unified Process. Advances in Cross-Language Information Retrieval. Results of the Cross-Language Evaluation Forum - CLEF 2002. Lecture Notes in Computer Science, Vol. 2785, Springer-Verlag, Germany (2003)
9. Braschler, M., Ripplinger, B. (2003). Stemming and Decompounding for German Text Retrieval. Advances in Information Retrieval. Lecture Notes in Computer Science, Vol. 2633. Springer-Verlag, Germany (2003) 177-192
10. Chen, A.: Multilingual Information Retrieval Using English and Chinese Queries. Evaluation of Cross-Language Information Retrieval Systems. Lecture Notes in Computer Science; Vol. 2406. Springer-Verlag, Germany (2002) 44-58
11. Nie, J.: Towards a unified approach to CLIR and multilingual IR. SIGIR 2002 Workshop I, Cross-language information retrieval: a research map. University of Tampere, Finland (2002) 8–14

Appendix

Example 1.
English – Finnish query no. 187 with the morphologically analyzed index
Average precision 100 %
#sum(#syn(ydin) #syn(kuljetus matkanaikana rahtimaksu kulkuneuvo pika kuljettaa) #syn(saksa) #syn(pitää jonakin löytää huomata löytö) #syn(todistus huhu pamaus ilmoittaa ilmoittautua) #syn(esittää vastalause vastalause paheksunta mielenosoitus rähinä vetoomus vastustaa kyseenalaistaminen) #syn(kuljetus) #syn(radioaktiivinen) #syn(tuhlata jäte haaskaus erämaa) #syn(pyörä majava majavannahka) #syn(astia kontti) #syn(saksa));

Example 2.
English – Finnish query no. 187 with the stemmed index
Average precision 16.7 %
#sum(#syn(yd) #syn(kuljetus matkan aik rahtimaksu kulkuneuvo pika kuljet) #syn(saks) #syn(löytä huoma pitää j löytö) #syn(todistus huhu pamaus ilmoit ilmoittautu) #syn(vastalaus paheksun mielenosoitus räh vetoomus vastust esittää vastalaus kyseenalaistamin) #syn(kuljetus) #syn(radioaktiivin) #syn(tuhl jäte haaskaus eräm) #syn(pyörä majav majavannahk) #syn(ast kont) #syn(saks));

Example 3.
English – Swedish query no. 141 with the morphologically analyzed index
Average precision 100.0 %
#sum(#syn(bokstav brev typ) #syn(bomb bomba) #syn(bluesbasera @bauer) #syn(komma på anse fynd) #syn(information) #syn(explosion utbrott spricka) #syn(bokstav brev typ) #syn(bomb bomba) #syn(studio) #syn(television tv tv-apparat tv) #syn(ränna segelränna kanal kanalisera) #syn(pro far förbivid proffs) #syn(7) #syn(lägga fram sätta upp höra upp presentera hallåa framlägga framföra) #syn(kabellag @arabella) #syn(bluesbasera @bauer));

Example 4.
English – Swedish query no. 141 with the stemmed index
Average precision 14.3 %

#sum(#syn(bokstav brev typ) #syn(bomb) #syn(bauer griesbaum)#syn(finn komma på ans fynd) #syn(information) #syn(explosion utbrot sprick) #syn(bokstav brev typ) #syn(bomb) #syn(studio) #syn(television tv tv-appar tv) #syn(ränn segelrän kanal kanaliser) #syn(pro far förbi, vid proff) #syn(7) #syn(presenter hallå framlägg lägga fram sätta upp höra upp) #syn(rabell larabell) #syn(bauer griesbaum));

Example 5.
English – German query no. 184 with the morphologically analyzed index
Average precision 67.5 %
#sum(#syn(mutterschaft) #syn(erlaubnis verlassen zurücklassen urlaub lassen überlassen hinterlassen) #syn(europa) #syn(finden feststellen fund) #syn(geben anrufen nachgeben nachgiebigkeit) #syn(information) #syn(versorgung vergütung vorkehrung vorrat bestimmung) #syn(betreffen beunruhigen beschäftigen angelegenheit sorge unternehmen) #syn(länge stück) #syn(mutterschaft) #syn(erlaubnis verlassen zurücklassen urlaub lassen überlassen hinterlassen) #syn(europa));

Example 6.
English – German query no. 184 with the stemmed index
Average precision 2.7 %
#sum(#syn(mutterschaft) #syn(erlaubnis verlass zurucklass urlaublass uberlass hinterlass) #syn(europ) #syn(find feststell fund) #syn(geb anruf nachgeb nachgieb) #syn(information) #syn(versorg vergut vorkehr vorrat bestimm) #syn(betreff beunruh beschaft angeleg sorg unternehm) #syn(stuck) #syn(mutterschaft) #syn(erlaubnis verlass zurucklass urlaub lass uberlass hinterlass) #syn(europ));

Example 7.
English – Finnish query no. 183 with the stemmed index
Average precision 0.0 %
#sum(#syn(aasialain) #syn(dinosaurus) #syn(täht maalliset jäännöks) #syn(jäädä jäädä ed) #syn(ran lohko puolue osuus ranniko hiekkaran äyräs rooli lävits ero) #syn(as tehtäv) #syn(dinosaurus) #syn(täht maalliset jäännöks) #syn(jäädä jäädä ed) #syn(perust perustu löytä huoma pitää j) #syn(löytä huoma pitää j löytö));

Example 8.
English – Swedish query no. 148 with the stemmed index
Average precision 4.7 %
#sum(#syn(skad skadestånd) #syn(frisk hav luft ozon störtskur) #syn(lag värphön) #syn(hål slå hål träffa hålet) #syn(frisk hav luft ozon störtskur) #syn(lag värphön) #syn(effek verkan åstadkomm) #syn (förorening)).

Study III

Eija Airio. 2006. Word normalization and decompounding in mono-and bilingual IR. Information Retrieval 9(3), 249-271.