*Post-print*

|  |  |
|---|---|
| Tekijät: | Airio Eija |
| Artikkelin nimi: | Word normalization and decompounding in mono- and bilingual IR |
| Julkaisuvuosi: | 2006 |
| Lehden nimi: | Information Retrieval |
| Lehden vol: | 9 |
| Lehden numero: | 3 |
| Sivunumerot: | 249-271 |
| ISSN: | 1386-4564 |
| Tieteenala: | Luonnontieteet / Tietojenkäsittelyoppi ja informaatiotieteet |
| Kieli: | en |

URN: http://urn.fi/urn:nbn:uta-3-473

Information Retrieval 9(3): 249-271

# Word Normalization and Decompounding in Mono- and Bilingual IR

EIJA AIRIO eija.airio@uta.fi
*Department of Information Studies, Kanslerinrinne 1, 33014 Tampere University, University of Tampere, Finland*

**Abstract.** The present research studies the impact of decompounding and two different word normalization methods, stemming and lemmatization, on monolingual and bilingual retrieval. The languages in the monolingual runs are English, Finnish, German and Swedish. The source language of the bilingual runs is English, and the target languages are Finnish, German and Swedish. In the monolingual runs, retrieval in a lemmatized compound index gives almost as good results as retrieval in a decompounded index, but in the bilingual runs differences are found: retrieval in a lemmatized decompounded index performs better than retrieval in a lemmatized compound index. The reason for the poorer performance of indexes without decompounding in bilingual retrieval is the difference between the source language and target languages: phrases are used in English, while compounds are used instead of phrases in Finnish, German and Swedish. No remarkable performance differences could be found between stemming and lemmatization.

Keywords: monolingual information retrieval, bilingual information retrieval, lemmatization, stemming, decompounding

## 1. Introduction

Word inflection is a feature of most natural languages. Verbs inflect according to the person, tense and possibly the finite form. Nouns inflect, among others, in the plural form, as well as in the case forms in some languages. The degree of inflection varies according to the language (see Pirkola 2001). Word inflection has its effect on information retrieval, because texts and queries include natural language words. It is possible that a query word and a word in a relevant document do not match because of inflection, although they would be inflected variants of the same basic word. In IR, various word form normalization methods have been developed to overcome problems produced by inflection. The normalization methods are applied both in text indexing and in retrieval. Word form normalization tools may be divided into two classes: stemmers and lemmatizers. Lemmatizers return a basic form of a word, the lemma, while stemmers return a string which is not inevitably any lexical word. The simplest stemming algorithms only strip off the word endings. Those algorithms perform “many – to – one” mapping, which means that distinct words may have an identical stem. (Koskenniemi 1983, 12).

English has quite simple word inflection rules, which has presumably had an impact on research: theoretical research in computational morphology has not interested IR researchers. However, in recent years, other languages than English have become popular in IR research. Especially cross-language IR (CLIR) research has exploited document collections in several different, and also morphologically complex, languages.

A range of varying types of indexes may be built for IR collections by employing distinct normalization methods. The present research analyzes the performance of retrieval in different index types on monolingual and bilingual IR. In a bilingual IR task, the source language and the target language differ, and the impact of the applied normalization method may differ from that of a monolingual task. It is vital for IR researchers to know, which normalization method performs best in IR and CLIR.

Previous research mostly shows that word normalization is advantageous in monolingual IR compared with inflected retrieval, especially in non-English retrieval (see Alkula 2000, Kettunen 2004, Popovic & Willet 1992, Braschler Ripplinger 2004 and Hollink & al. 2004). In highly inflectional languages, for example Finnish, German and Slovene, normalization is without exception advantageous (see Section 4). In English retrieval, the importance of normalization is not so evident (see Harman 1991, Popovic & Willet 1992 and Krovetz 1993). Stemming has been the most common normalization method in IR tests.

The present research studies the impact of two different word normalization methods, stemming and lemmatization, as well as the impact of decompounding (called also compound splitting), on monolingual and bilingual retrieval. In the monolingual tests, retrieval without normalization is the baseline, and in bilingual tests, the baseline is retrieval with lemmatization and decompounding. Lemmatization has not been studied sufficiently in previous research.

There are four test languages in this research: English, Finnish, Swedish and German. English is the source language in the bilingual tests. CLEF 2003 topics and datasets are utilized. In the bilingual tests, the dictionary-based approach is used.

The structure of this paper is the following. Section 2 presents the main methods used in cross-language information retrieval. Section 3 describes the word normalization methods used in IR, and Section 4 discusses their effects on retrieval result. Resources, data and methods of present study are discussed in Section 5. Section 6 presents the runs and the results, and Section 7 contains the discussion. Finally, conclusions are presented in Section 8.

## 2. CLIR approaches

Bilingual information retrieval is a subset of cross-language information retrieval, CLIR. There are two main CLIR approaches: either to translate the queries into the document language(s), or to translate the documents into the source language. The first one is easier and cheaper to carry out than the latter, and it is applied in the present study as well.

There are various translation approaches for CLIR. An easy approach is to use **machine translation** (MT). MT resources are not available for all languages pairs, however, and they are very expensive. MT systems give only one translation variant for each source word, which is not advantageous for IR purposes. The **dictionary-based approach**, which is used in this study, has not the disadvantages of the MT approach. There are many free or low-cost machine-readable dictionaries. Translation dictionaries give several translations for a word, which is advantageous for IR. (Kraaij 2004, 124-125.)

In MT systems and translation dictionaries, only basic word forms match the dictionary. In order to match query words with dictionary words, query words must be lemmatized. (Kraaij 2004, 124.) In the dictionary-based approach, stemming or n-gramming query words and dictionary words is possible as well. MT systems and dictionaries give only lemmas as their output. The same type of normalization tool (a lemmatizer or a stemmer) must be used for indexing the database and query word normalization. An alternative way is to use n-grams in indexing and query word processing (McNamee & Mayfield 2001).

In the **corpus-based approach** a probabilistic dictionary is derived from parallel corpora. The simplest approach is to assume one-to-one mapping between words. It is often not a reasonable approach, however, because the word order may vary between languages. Sentence alignment is a more widely used approach than word alignment. Corpora are usually domain dependent, which can be a drawback or an advantage: probabilistic dictionaries derived from parallel corpora often are narrow, but they may be suitable for translating special terminology. (Kraaij 2004, 125.) Dictionaries derived from parallel corpora are not restricted to lemmas, but may include inflected word forms as well. Thus it is possible to utilize them without lemmatizing source words. Retrieval may be performed in inflected word form index using translations as such, or translated words may be normalized and retrieval performed in a normalized index.

## 3. Word normalization tools

The two groups of word normalization tools, stemmers and lemmatizers, are not distinct, because some normalization tools may be categorised to one or the other group, depending on the definition. We use the definition **lemmatizer** here for a normalization tool, which returns the basic forms of a word, **lemma**, and utilizes morphological rules. By **stemmer** we refer to a normalization tool, which returns a “stem” for each word. The stem is not necessarily any real word of a language, but it may be for example a truncated form of a word, when a basic form of a word is a lexical word.

### *3.1. Stemmers*

There are various stemming strategies developed for different purposes. Some stemming algorithms utilize a stem dictionary and others a suffix list. Many stemming algorithms, whose purpose is to improve IR performance, do not use a stem dictionary, but an explicit list of suffixes, and the criteria for removing suffixes. Stemmers of the perhaps most popular stemmer family today, the Porter stemmers, have adopted this approach. (Porter 1980.)

When developing a suffix stripping algorithm for IR, the main goal is to improve IR performance, not to follow linguistically authentic rules. Porter gives criteria for stemming two words to a single stem: if there is no difference between the two statements ‘a document is about W1’ and ‘a document is about W2’, W1 and W2 may be stemmed to a single stem. However, there is often some variation in opinion concerning the two words W1 and W2, and thus the decision whether they should be conflated or not is not so clear. (Porter 1980.)

The very first stemmers were simple: they just stripped off the endings. For example Lovins created principles for developing stemming algorithms in 1969 (Koskenniemi 1983, 12). The idea of stemming may be illustrated by giving the sample *connect, connected, connecting, connection and connections*. These words have a similar meaning, and it would be reasonable to stem them to a common form. If suffixes *ed*, *ing*, *ion* and *ions* are removed, the stem will be *connect* for all these words. (Porter 1980.)

When the stemmer removes too small a suffix, we speak about **under-stemming**. Under-stemming is for example removing the suffix *s* from the word *babies*. **Over-stemming** is the opposite of under-stemming: the stemmer removes too long a suffix. An example of over-stemming could be stemming the English word *probably* to a stem *prob*. Porter presents the term **mis-stemming** in addition to under-stemming and over-stemming. 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 taking off the suffix *ly* from an English word is right in most cases, e.g. *cheaply*, but it should not be taken off from the word *reply*. (Porter 1981).

There are various results concerning the effect of stemmers on the performance of IR tasks. Kraaij (1996) reviewed research on stemmers in IR. He found that many factors affect the result. Linguistic vs. non-linguistic stemmers, various languages, and varying query and document length all have an impact on retrieval results. (Kraaij 1996, 41).

### *3.2. Lemmatizers*

Lemmatizers utilize lexica to recognize all possible lexical representations of word-forms. Rules are needed to express the permitted relations between lexical and surface representations (the surface representation refers to the appearance of the word in the text).

Niedermair and others use a morpheme-dictionary, a morpheme-grammar, and a decomposition automaton for lemmatization. The morpheme-dictionary contains word-stem affixes, inflectional endings and fillers. The morpheme-grammar splits the word into its prefix-, stem-, derivational-, and inflectional elements. They are represented in a uniform way. (Niedermair & al. 1984, 375-377.)

Koskenniemi describes a two-level model of a lemmatizer, which has two major components: a lexicon system and a collection of rules. The rules define how affixes may be joined to words. The model is language independent: new languages may be introduced by describing the lexicon and rules of a language. (Koskenniemi 1985, 1-2) The lemmatizers applied in this study (FINTWOL, ENGTWOL, GERTWOL and SWETWOL) are based on this model. The TWOL –lemmatizers give all interpretable basic forms for a word, as well as word class, case etc. For example for the word *saw* ENGTWOL gives two interpretations:

"saw"
"see"

where *saw* is interpreted as a nominal, and as the past tense of the verb *see*.

Lemmatizers are capable of splitting compounds into their constituents, and of lemmatizing parts of the compound as well. For example, FINTWOL gives the following reading for the Finnish word *tiedonhaku* (information retrieval):

"tiedonhaku"
"tieto#haku"

where FINTWOL recognizes the compound *tiedonhaku* as well as its parts: *tieto* (information) and *haku* (retrieval) in their lexical form.

Alkula (2000) calls **parasite words** strings which are correct word forms as such, but which are not real constituents of the current word. A lemmatizer may find parasite words in some cases: it may misinterpret constituents of a word. The Swedish lemmatizer SWETWOL gives following interpretations for a word *bilimport* (car import):

"bil#import"
"bi#lim#port"
"bi#limpa#ort"

Only the first interpretation of SWETWOL is correct: bilimport has two parts: *bil* (a car) and *import* (import). The two other interpretations: *bi* (subsidiary), *lim* (paste), *port* (a door) and *bi*, *limpa* (a loaf) and *ort* (a locality), are wrong interpretations consisting of parasite words.

In some situations, the way the lemmatizer functions is not appropriate for IR. In IR, we are interested in any reasonable interpretation, not the basic form, of the word. Sometimes these are the same thing, but not always. For example, it would be reasonable to index English words *connect* and *connection* in a common entry. A lemmatizer, however, gives different basic forms (*connect* and *connection*) for those words.

The lemmatizer performs faultlessly in this situation from the linguistic viewpoint. From the IR point of view, reducing both to a common form *connect* would be a better solution. However, that would not be lemmatization anymore.

## 4. The effects of normalization tools and decompounding on retrieval results

There are several studies on the effect of various word normalization methods on IR results. Most of the studies compare the performance of an inflected word form index and inflected queries with the performance of a stemmed index and stemmed queries. Next, the studies on English retrieval are summarized separately of the studies on non-English retrieval, because their results are different.

### *4.1. Monolingual English tests*

Stemmers are more widely applied in IR-tests than lemmatizers, and English has been the most common test language. Harman studied the interaction of stemmers and ranking techniques in retrieval performance. She tested three general purpose stemmers. None of the stemmers achieved any further improvement over term weighting approach, where query words were in inflected form. (Harman 1991, 9). Hull criticised Harman's conclusions concerning poor performance of retrieval in a stemmed index. He stated that stemming is almost always beneficial. According to Hull, Harman's conclusions were different from his, even if the results of both were quite similar. There are two reasons for that. First, Harman used full TREC queries in her tests. When shorter queries are used, stemmed queries always outperform inflected ones. Second, Harman used cutoffs of 10 and 30 documents, which, according to Hull, are not large enough in large collections. (Hull 1996, 83.)

Popovic and Willet compared performance of retrieval in an inflected English index with performance of retrieval in a linguistically stemmed English index. They did not find any statistically significant performance differences. (Popovic & Willet 1992, 390.) Lennon and others evaluated the impact of several stemmers on IR result. Even if the tested stemmers were developed separately and based on different principles, only minor differences in retrieval performance were found. (Lennon & al, 1981, 177.) Krovetz compared the performance of retrieval in an English stemmed index and in an English inflected word form index. He found significant improvements in performance of the first one compared with the latter. (Krovetz 1993, 202.)

The effect of lemmatization on the English retrieval results has not been studied widely. Niedermair and others found that recall increased 68% with their MARS lemmatizer compared to the recall without MARS, bur precision dropped form 68% to 61%. (Niedermair & al. 1984, 379.)

Most of the studies which compare retrieval performance in an inflected word form English index with that of a normalized index show only minor improvement for the latter. Comparing the results described above, there have been only few controversial results.

### *4.2. Monolingual non-English tests*

Alkula has compared the retrieval performance in an inflected word form index and in a lemmatized index in monolingual Finnish retrieval. She found that the precision of runs in a lemmatized index was mainly higher than that of runs in an inflected word form index. The lemmatized index with lemmatized, decompounded queries, and the inflected word form index with automatic truncated queries gave the best precision. The latter, however, obtained the poorest recall ratio. (Alkula 2000, 7-8.)

Kettunen and colleagues have tested performance of stemming and lemmatizing in monolingual Finnish retrieval. The decompounded index was utilized in the lemmatized runs. Both methods achieved better results than the baseline, which was retrieval in inflected word form index: the average precision of the inflected word form run was 18.9 %, while it was 27.7 % in the stemmed run and 35.0 % in the lemmatized run. (Kettunen & al. 2004.)

Popovic and Willet tested the effect of stemming on Slovene IR. They found that the retrieval results with an appropriate stemmer are statistically better than the results without stemming. (Popovic & Willet 1992, 390.)

Braschler and Ripplinger studied stemming and decompounding in German monolingual retrieval. They compared the retrieval results utilizing different word normalization methods with each other. The normalization methods tested were a combination of word-based and n-gram based retrieval, automatic machine learning, the NIST stemmer, the Spider stemmer and morpho-syntactic analysis. The run without normalization (an inflected word form index) was the baseline. The authors found that all the runs utilizing a normalization method outperformed the inflected form run. The run with a combination of word-based and n-gram based retrieval was the worst of the normalized runs, but there were no other large differences between the runs. (Braschler & Ripplinger 2004, 295-306.)

Hollink and colleagues (2004) investigated monolingual retrieval in several European languages. They compared the performance of inflected word form indexes with the performance of stemmed indexes. They found that stemming improves the results but depends on the language: the highest increase was attained in Finnish, where the result with the stemmed index was 30 % better than that with the inflected word form index. In Dutch and French, the result with the stemmed index was only 1,2 % better than with the inflected word form index. (Hollink & al. 2004, 36-37.)

Larkey and colleagues (2002) developed several light stemmers for Arabic. The stemmers were very simple, and did not take into account most of Arabic morphology. The authors compared the results given by the light stemmers with the result given by a morphological stemmer which tries to find the root for each word (and thus performs analogously with a lemmatizer). The authors found that one of the light stemmers achieved the best result, around 100 % increase in average precision from raw retrieval. The best light stemmer outperformed the morphological stemmer as well. The authors conclude that it is probably not essential for a stemmer to yield the correct forms, but to group most of the forms that belong together. (Larkey & al. 2002, 275-280.)

Word normalizing seems to be mostly advantageous in non-English retrieval. However, Hollink and colleagues found that stemming may be advantageous for some languages but not for all inside a single language family. (Hollink & al. 2004, 38-39.) The quality of the stemmers might be one possible reason for these differences.

### *4.3. Bilingual tests*

In bilingual dictionary-based IR, inflected retrieval is not practical (unless special methods, for example n-gramming, is utlized), because dictionaries usually include target words in their basic forms. The only reasonable alternatives are stemming or lemmatization. Probabilistic dictionaries, derived from parallel corpora, include inflected word forms. When they are utilized, inflected retrieval is sensible.

Larkey and colleagues tested the performance of their light stemmers on bilingual retrieval, and compared the results with the performance of the morphological stemmer (see Section 4.2). One of the light stemmers achieved the best result, and outperformed the result of the morphological stemmer. (Larkey & al. 2002, 280-281.)

## 5. Resources, data and methods

### *5.1. Research questions*

According to previous research, word form normalization has an advantageous effect on the result of monolingual non-English retrieval. The results concerning English retrieval are not so clear: in several tests normalization has not had any notable impact, but there are controversial results as well. Effects of the two word form normalization methods, stemming and lemmatizing, have not been compared widely in previous IR research. This concerns as well monolingual as bilingual IR. These observations give rise to the following research questions:

1. Does monolingual retrieval with normalization give significantly better results than retrieval without normalization?
2. Which gives better results in monolingual runs, retrieval with stemming in the stemmed index, retrieval with lemmatization in the lemmatized compound index or retrieval with lemmatization in the lemmatized decompounded index?
3. Which gives better results in bilingual runs, retrieval with stemming in the stemmed index, retrieval with lemmatization in the lemmatized compound index or retrieval with lemmatization in the lemmatized decompounded index?

### *5.2. Language resources and collections*

In this section, we describe the language resources and collections used in this research.

The following language resources were used in the tests:

- Motcom GlobalDix multilingual translation dictionary (18 languages, total number of words 665 000, 44 000 English entries, 26 000 Finnish entries, 39 000 German entries, 36 000 Swedish entries) by Kielikone plc. Finland
- Lemmatizers FINTWOL GERTWOL, SWETWOL and ENGTWOL by Lingsoft plc. Finland
- Stemmers for English, German, Finnish and Swedish, SNOWBALL stemmers by Martin Porter
- English stop word list (429 stopwords), created on the basis of InQuery's default stop list for English
- Finnish stop word list (773 stopwords), created on the basis of the English stop list
- Swedish stop word list (499 stopwords), created at the University of Tampere (UTA)
- German stop word list (1318 stopwords), created on the basis of the English stop list

The lemmatizers used in the tests are based on a two-level model. They give all the possible base forms for a given inflected word and are capable of splitting compounds. The Snowball stemmers used in the tests are algorithmic and simple. They do not utilize any dictionaries or exception lists. (Porter 1981).

CLEF 2003 datasets (English, Finnish, German and Swedish) were used for the tests (see Table 1).

**Table 1.** CLEF 2003 datasets.

| **Collection language** | **Source** | **Number of documents** | **Size of the corpus (MB)** |
|---|---|---|---|
| English | Los Angeles Times 1994<br>Glasgow Herald 1995 | 169,477 | 579 |
| Finnish | Aamulehti 1994-1995 | 55,344 | 137 |
| German | Rundschau 1994<br>Der Spiegel 1994-1995<br>SDA German 1994-1995 | 294,809 | 668 |
| Swedish | Tidningarnas Telegrambyrå 1994-1995 | 142,819 | 352 |

We utilized CLEF 2003 topics and relevance assessments in the tests. There are 60 CLEF 2003 topics, translated into all the CLEF languages, including the present test languages.

The *InQuery* system, provided by the Center for Intelligent Information Retrieval at the University of Massachusetts, was utilized in indexing the databases and as the retrieval system. The stemmers and the lemmatizers were utilized in indexing, as well as in pre-processing and post-processing of query words.

### *5.3. Indexes, translation approach and runs*

The aim of this research is to compare performance of different word normalization tools and decompounding in monolingual and bilingual IR. For that purpose, four kinds of indexes were created: inflected, stemmed, lemmatized with decompounding and lemmatized without decompounding. Altogether 15 indexes were created: four Finnish, German and Swedish indexes (inflected, stemmed, lemmatized decompounded and lemmatized compound index), and three English indexes (inflected, stemmed and lemmatized compound index). For English, no decompounded index was created, because of lack of decompounding tools for English. On the other hand, compounds are quite rare in English, and do presumably not constitute any great difficulties in retrieval.

The word tokenization rules used in indexing were following. First, punctuation marks were deleted. Next, strings broken down by the space character were decoded to be indexable words. Capitals were converted into lower case letters before indexing.

Altogether 24 test runs were performed, out of which 15 were monolingual and 9 bilingual. The languages of the monolingual runs were English, Finnish, German and Swedish. Inflected, lemmatized (in the compound index) and stemmed runs were performed for all four languages, and in addition lemmatized runs in decompounded index for Finnish, German and Swedish. The source language of all the bilingual runs was English, and the target languages were Finnish, German and Swedish. Two lemmatized runs (one in the decompounded index and one in the compound index) and one stemmed run were performed for all the language pairs.

The UTACLIR query translation system of University of Tampere was used in the test. The system utilizes external language resources (translation dictionaries, stemmers and lemmatizers). Word processing in UTACLIR proceeds as follows. First topic words are normalized with a lemmatizer. The existence of a lemmatizer for the source language is vital, because stemmed words do not match lemmas in the dictionary. The lemmatizer produces one or more basic forms for a token. After normalization, stop-words are removed, and non-stop words are translated. If translation equivalents are found, they are normalized utilizing a lemmatizer or a stemmer, depending on the target index. Queries are structured utilizing a synonym operator (see Pirkola 1998): the target words derived from the same source word are grouped into the same synonym group. (Airio & al. 2003, 92-93.)

The UTACLIR approach handles distinct source words: we have no phrase recognition for the source language. This solution is based on our assumption that our translation dictionary contains only few phrases and compounds. On the other hand we assume that many phrases, as well as compounds, present in documents and queries are not customary, but are composed contemporarily.

In our *bilingual runs*, the query words were first normalized utilizing the English lemmatizer. In the *bilingual stemmed runs*, translations were normalized utilizing the stemmer, and retrieval was performed in the stemmed index. The lemmatizer was utilized for word normalization in the *bilingual lemmatized runs*, and retrieval was performed in either of the lemmatized indexes.

The approach in the *monolingual stemmed runs* was to stem the topic words, and perform retrieval in the stemmed index. In the monolingual *lemmatized runs*, the topic words were lemmatized, and retrieval was performed in either of the lemmatized indexes.

In the *inflected word form* runs, topic words were added as such into the query, and retrieval was performed in the inflected word form index.

## 6. Results

*6.1. Bilingual runs*

Our supposition concerning phrases in English topics seemed to be correct: we found 42 phrases among the topics, out of which one (fast food) could be translated utilizing our translation dictionary. The reason for the high number untranslatable phrases might be partly the quality of the dictionary: among those 42 phrases, we found seven customary ones (for example *mobile phone*), which should be included in the dictionary. But the rest, 35 phrases, were more or less contemporary ones (for example *diamond industry*, *purple cabinet*, *flood disaster*), which cannot be assumed to be included in a standard translation dictionary.

Retrieval in the lemmatized indexes where compounds were split performed best in all the bilingual runs. In English-Finnish and English-German runs, the next best was the run in the lemmatized compound index, and the stemmed run achieved the worst result (see Table 2 and Figures 1, 2 and 3). In the English-German run, the difference between the result of the run in the lemmatized compound index and the result of the stemmed run was only minor: the stemmed run performed only 2.7 % worse than the run in the lemmatized compound index. In the English-Finnish run, the stemmed run performed clearly worse than either of the lemmatized runs: the result was 41.4 % worse than that of the lemmatized decompounded index, and 28.3 % worse than the result of the run in the lemmatized compound index.

In the English-Swedish runs and in the English-German runs, the differences between the two lemmatized runs were statistically significant by the Wilcoxon signed ranks test at the 0.01 level, but differences between the run in the lemmatized compound index and stemmed run were not significant. In the English-Finnish run the situation is opposite: the differences between the two lemmatized runs were not statistically significant, but between the run in the lemmatized compound index and stemmed run they were significant.

All the differences between the bilingual stemmed runs and the runs in the lemmatized decompounded indexes were statistically significant by the Wilcoxon signed ranks test at the 0.01 level.

**Table 2.** 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. nosplit run)** | **Change % (from the lemm. nosplit run** |
|---|---|---|---|---|---|---|
| Finnish | lemmatized+split | 35.5 | | | | |
| Finnish | lemmatized, nosplit | 29.0 | -6.5 | -18.3 | | |
| Finnish | stemmed | 20.8 | -14.7 | -41.4 | -8.2 | -28.3 |
| Swedish | lemmatized+split | 27.1 | | | | |
| Swedish | lemmatized, nosplit | 17.4 | -9.7 | -35.8 | | |
| Swedish | stemmed | 19.0 | -8.1 | -29.9 | 1.6 | 9.2 |
| German | lemmatized+split | 31.0 | | | | |
| German | lemmatized, nosplit | 26.4 | -4.6 | -14.8 | | |
| German | stemmed | 25.7 | -5.3 | -17.1 | -0.7 | -2.7 |

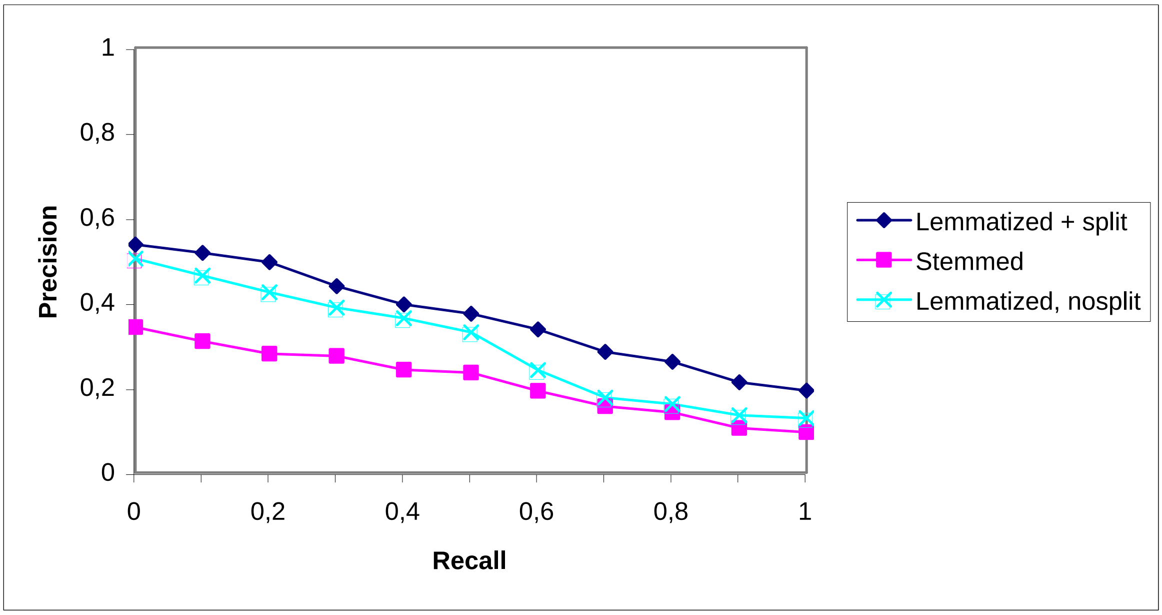

**Figure 1.** PR-curves for English - Finnish runs.

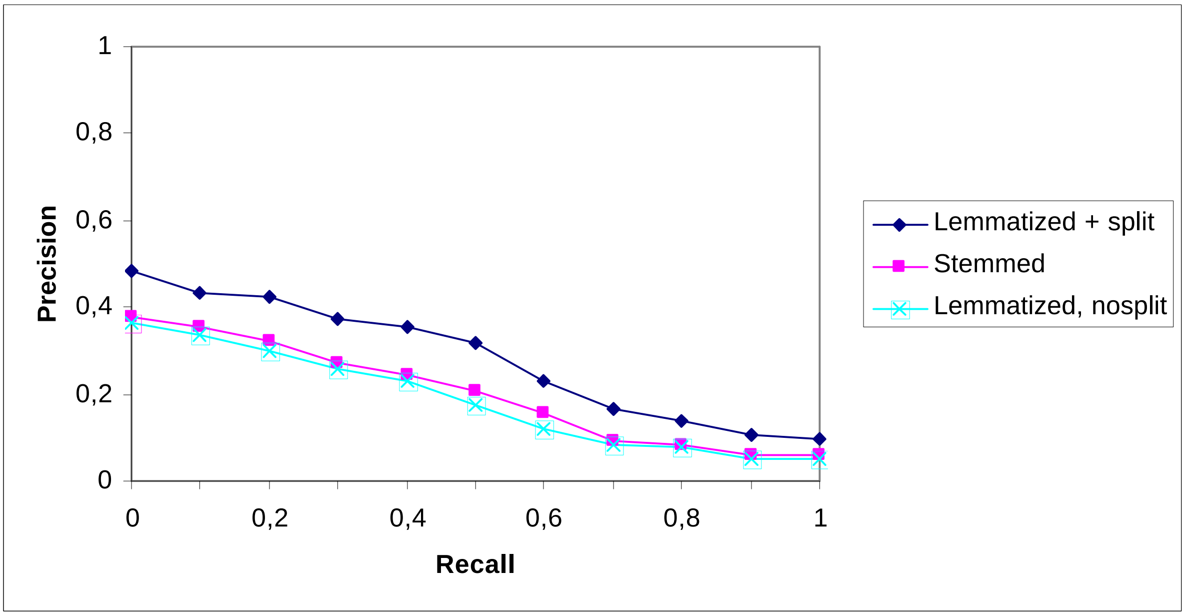

**Figure 2.** PR-curves for English - Swedish runs.

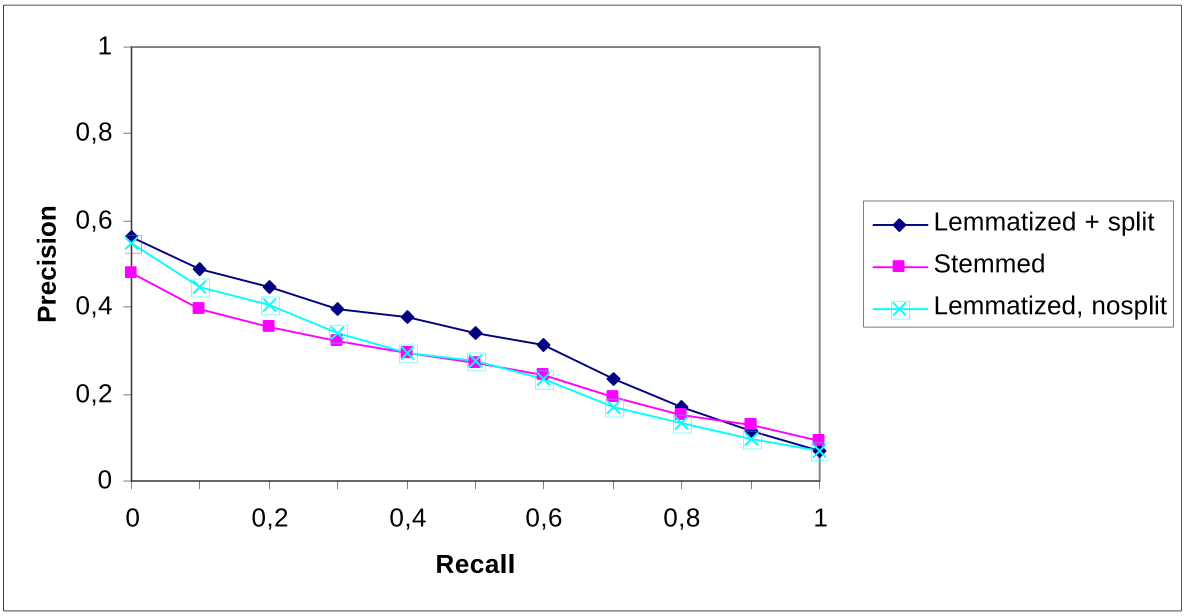

**Figure 3.** PR-curves for English - German runs.

Next, individual queries of bilingual runs are analyzed more closely to detect the reasons for the performance differences between the two normalization methods and decompounding. The selection of examples was performed in the following way. First, we identified clear performance differences between the runs concerning individual topics. We found that performance of various methods differed topic by topic: the run which achieved the worst average precision, could achieve the best result in a single topic. We analyzed the translated queries, as well as the source topic, to find out the reasons for the differences. In some cases, we analyzed also about 10 first document matches for the queries to find out possible the problematic query words.

*Compounds*

One possible source of problems in the bilingual stemmed runs and the runs in the lemmatized compound index is the handling of compounds (see Hedlund & al. 2002a, 127-128). English, which is the source language in our bilingual runs, includes a lot of phrases: two or more words composing a new word are written separately. Words *information* and *retrieval* written sequentially compose a phrase *information retrieval*. In Finnish, German and Swedish compounds are used instead of phrases. In a compound, the parts of the new word are written together. The compound parts may occur in the base form or in the inflected form. In Finnish the concept *information retrieval* is composed of the words *tieto* (information) and *haku* (retrieval). The first word is inflected, and the words are written together to form a compound *tiedonhaku*. In Swedish and German the compounds are composed in similar way as in Finnish, and they have joining morphemes. (Hedlund & al. 2001, 153-154.)

When the source language of a bilingual IR task is a language using phrases, and compounds are used in the target language instead, problems may occur. Words composing a phrase are translated independently. For the English phrase *information retrieval*, we would get Finnish translation *tieto haku*. A *lemmatized decompounded* index contains the whole compound as well as parts in their base form. For example, the Finnish lemmatized decompounded index contains words *tieto*, *haku* and *tiedonhaku* for the token *tiedonhaku*. In this case, compounds do not cause problems. In our example, translated topic words *tieto* and *haku* are further normalized with a lemmatizer, producing basic forms (identical to input words) *tieto* and *haku*, which match the appropriate index words.

If no compound splitting is performed in indexing, only the full compound is in the index, not its parts. This causes problems, because the query includes only parts of the compounds. For example, the Finnish stemmed index contains the stem *tiedonhaku* for the token *tiedonhaku*. For the topic phrase *information retrieval*, the translated and stemmed query contains strings *tied* and *haku,* which do not match the index.

Compounds caused problems in all our bilingual runs. In English – German and English – Swedish runs, the weaker performance of the stemmed run was caused mostly by problematic compounds: the lemmatizer without decompounding performed almost equally as the stemmer. Below we consider some examples of compound problems.

*English – Finnish*. English topic number 187 includes a phrase *nuclear transport*. The parts of this phrase are translated independently. The corresponding compound in Finnish is *ydinjätekuljetus*. When *decompounding* is not applied in indexing, we have only the compound *ydinjätekuljetus* in normalized form in the index. No matches are found in retrieval. When indexing is performed utilizing the *lemmatizer with decompounding*, parts of the phrases match with the parts of the compound. The weaker performance of the stemmed index compared to the performance of the lemmatized compound index is due to under and over-stemming cases. See Examples 1 and 2 in the Appendix.

*English – Swedish*. The same phenomenon, problems in compound splitting, can be seen in topic 186 in English – Swedish runs. The phrase in the topic is *purple cabinet*, and the translated query includes Swedish variants for those words, *purpur* and *koalition*. The corresponding Swedish compound is *purpurkoalition*. The index without decompounding includes the compound in the normalized form, while the decompounded index includes the full compound as well as its components in their normalized forms. Now parts of the phrase in the query match with the parts of the compound in the lemmatized decompounded index, but no matches are found in the stemmed index or in the lemmatized compound index. As in the previous example, the weaker performance of the stemmed index compared with the performance of the lemmatized compound index is due to under and over-stemming cases. See Examples 3 and 4 in the Appendix.

*English – German*. The English topic 184 includes a 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 index without decompounding includes only the compound *Mutterschaftsurlaub* in its normalized form, but the decompounded index includes the parts of the compound as well. See Examples 5 and 6 in the Appendix.

*Over-stemming*

Over-stemming happens, when too long a suffix is removed from the word. Then two or more words with separate meaning may get the same stem, which contributes to loss of precision. In our tests, over-stemming happened mostly in the stemmed English – Finnish run, where four queries were clearly affected. This may be considered as a quality issue of the Finnish SNOWBALL stemmer as well as an indication of complexity of Finnish.

*English – Finnish*. The topic 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 that also the word *tähti* (a star) has the same stem, which causes noise in retrieval. The average precision of this topic is 0.0 % in the stemmed run, while it is 50 % and 66.7 % in the runs with the lemmatized index. See Example 7 and 8 in the Appendix.

*Under-stemming*

Under-stemming occurs, when the suffix removed from the word is too short. Then words with the same meaning get separate stem, which contributes to loss of recall. Clear under-stemming cases could be found in the bilingual stemmed English – Finnish run only. There were under-stemmed words in every query in the English-Finnish run, but most of them did not have any impact on query performance.

*English – Finnish*. Topic 174 includes twice the word *Bavarian*, which is translated into Finnish as *baijerilainen*, and further stemmed into a stem *baijerilain*. In relevant documents the Finnish word *baijerilainen* occurs in inflected forms as well: *baijerilaisen*, *baijerilaisten* etc. The stemmer does not give the same stem for these inflected forms which it gives for the basic form, however. For *baijerilaisen* the stemmed form is *baijerilais* and for *baijerilaisten* it is *baijerilaist*. See Examples 9 and 10 in the Appendix.

*6.2. Monolingual runs*

All the monolingual non-English normalized runs performed better than the inflected runs (see Table 3 and Figures 4, 5, 6, and 7). The differences are statistically significant by the Wilcoxon signed ranks test at the 0.01 level. The only exception was the run in the German lemmatized compound index, whose result did not differ significantly from the baseline. The results of English monolingual runs are in line with the majority of the earlier results: no statistically significant differences could be found between the inflected run and the normalized runs.

In all the non-English runs, 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. The largest difference between the results of different indexing methods can be found in monolingual Swedish runs, where retrieval in the lemmatized compound index performed 19.1 % worse than retrieval in the lemmatized decompounded index.

The differences between the Swedish run in the lemmatized decompounded index and in the stemmed index are statistically significant by the Wilcoxon signed ranks test at the 0.01 level. In the German monolingual runs, the differences between the run in the lemmatized decompounded and the run in the lemmatized compound index are statistically significant. There are no statistically significant differences between the runs with various normalization types in other test languages.

**Table 3.** Non-interpolated average precision of monolingual runs for all relevant documents averaged over queries.

| Language | Index type | Average prec. % | Differ. % (from the baseline) | Change % (from the baseline) | Differ. % (from the lemm.split. run) | Change % (from the lemm.split. run) |
|---|---|---|---|---|---|---|
| English | inflected | 43.4 | | | | |
| English | lemmatized,nosplit | 45.6 | +2.2 | +5.1 | | |
| English | stemmed | 46.3 | +2.9 | +6.7 | +0.7 | +1.5 |
| Finnish | inflected | 31.0 | | | | |
| Finnish | lemmatized+split | 50.5 | +19.5 | +62.9 | | |
| Finnish | lemmatized,nosplit | 47.0 | +16.0 | +51.6 | -3.5 | -7.0 |
| Finnish | stemmed | 48.5 | +17.5 | +56.5 | -2.0 | -4.0 |
| Swedish | inflected | 30.2 | | | | |
| Swedish | lemmatized+split | 38.8 | +8.6 | +28.5 | | |
| Swedish | lemmatized,nosplit | 31.4 | +1.2 | +4.0 | -7.4 | -19.1 |
| Swedish | stemmed | 33.5 | +3.3 | +10.9 | -5.3 | -13.7 |
| German | inflected | 30.2 | | | | |
| German | lemmatized+split | 36.2 | +6.0 | +19.9 | | |
| German | lemmatized,nosplit | 31.9 | +1.7 | +5.6 | -4.3 | -11.9 |
| German | stemmed | 35.7 | +5.5 | +18.2 | -0.5 | -1.4 |

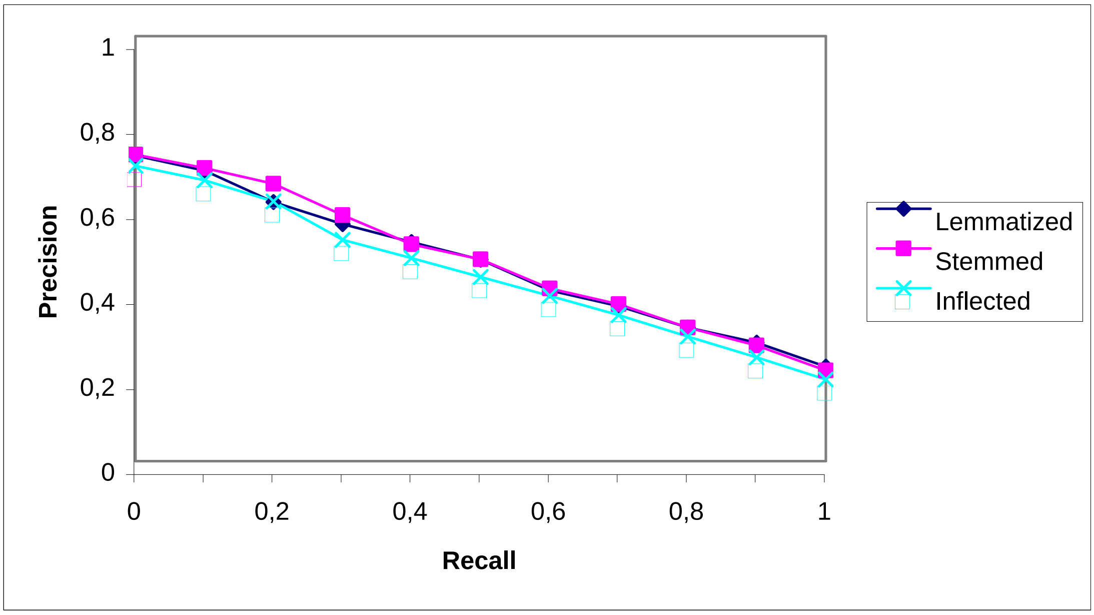

**Figure 4.** PR-curves for monolingual English runs.

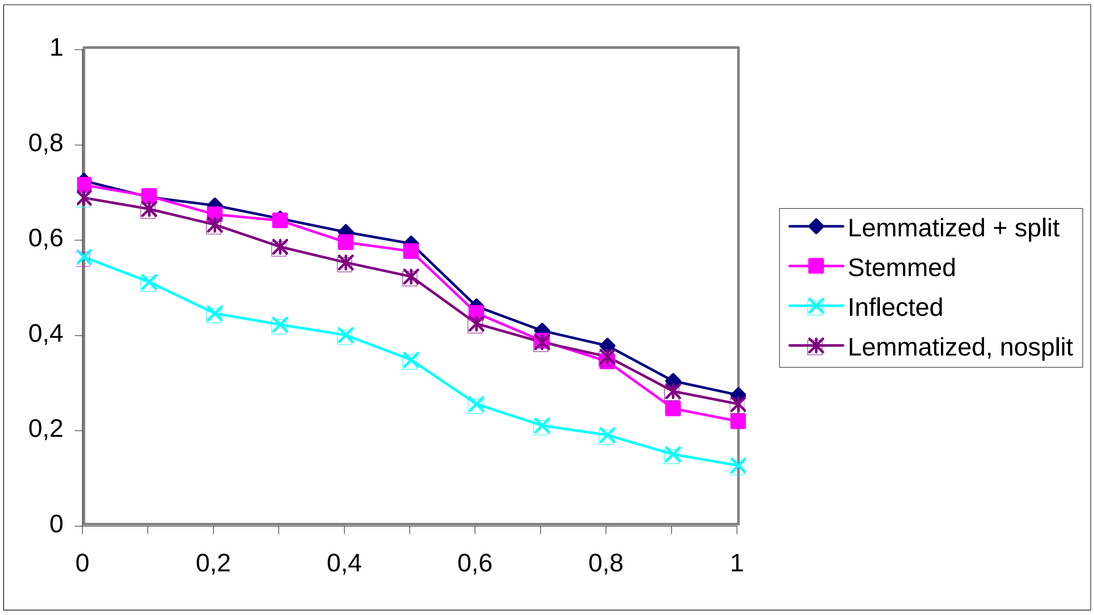

**Figure 5.** PR-curves for monolingual Finnish runs.

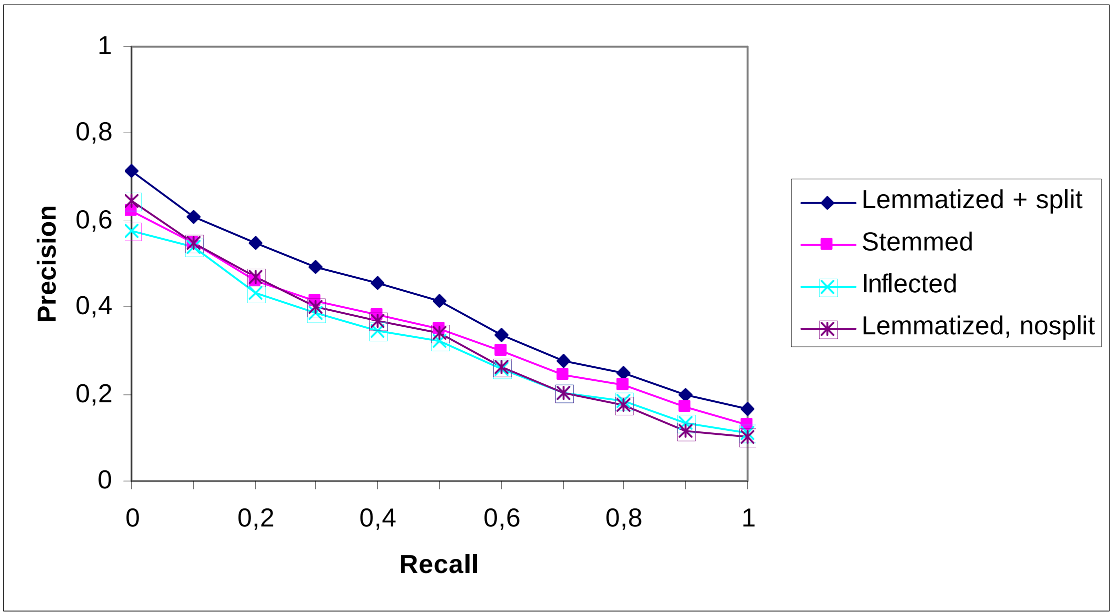

**Figure 6.** PR-curves for monolingual Swedish runs.

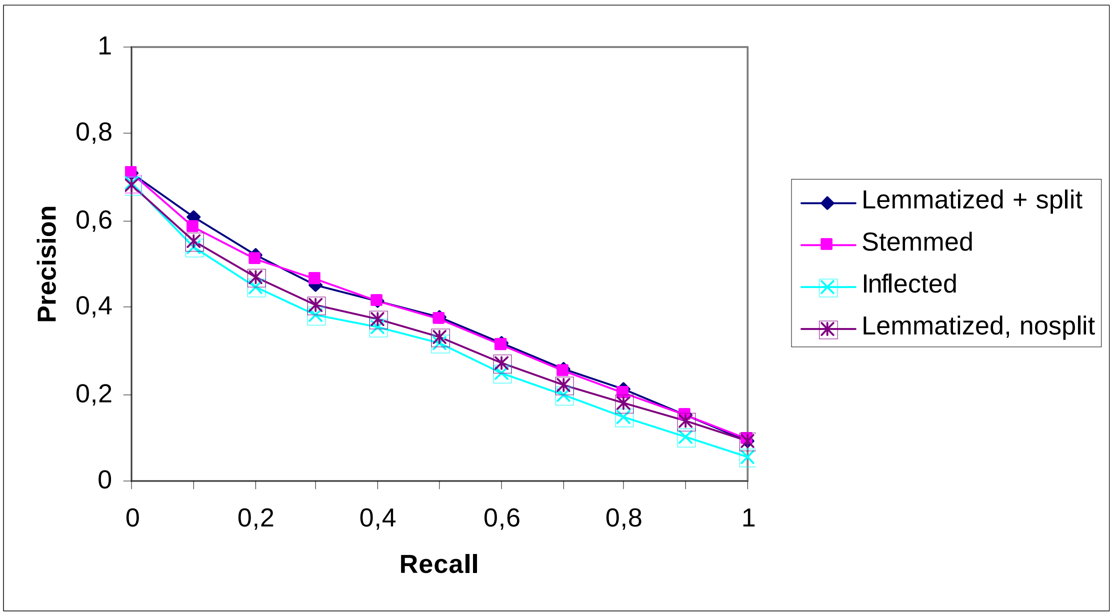

**Figure 7.** PR-curves for monolingual German runs.

Next, individual queries of monolingual runs are analyzed closer to detect the reasons for the performance differences of individual topics between the two normalization methods and decompounding.

*Compounds*

In the previous section, we concluded that problems with compounds occur in the bilingual runs when phrases are used in the source language, while the target language uses compounds. There is no reason, why compounds should cause problems in monolingual runs. The results of our monolingual runs seem to support this conclusion. Closer analysis of individual queries shows that the decompounded index may give better the results in queries containing compounds. Compound splitting in indexing phase acts like query expansion in retrieval phase. The reason for outperformance of the Swedish run in the lemmatized decompounded index compared with the stemmed run and the run in the lemmatized compound index seems to be mostly due to compound splitting.

*Finnish monolingual*. The Finnish topic 147 includes word *lintu* (a bird). In some relevant documents, the word *bird* occurs only as a part of a compound: *lintuparvi* (a flock of birds) or *lintuvahinko* (a bird accident). Again, the lemmatized decompounded index includes the compound itself (*lintuparvi*), as well as parts of the compound in a normalized form (*lintu* and *parvi*). Thus, the topic word *lintu* matches the decompounded index, but not the stemmed index nor the lemmatized compound index. The reason for the weaker performance of the lemmatized compound index compared to the stemmed can be explained by under-stemming, which happens to be advantageous in this topic. See Examples 11 and 12 in the Appendix.

*Swedish monolingual*. The Swedish topic 197 includes the word *Dayton*. Relevant documents include compounds *Dayton-samtal* and *Dayton-samtalet*, which can be found with the query word *dayton* in the decompounded index, but not in the compound index. The better result of the stemmed index compared with both lemmatized indexes is due to unrecognized words, which is explained below. See Examples 13 and 14 in the Appendix.

*Under-stemming*

*Finnish monolingual*. The Finnish topic 152 is a good example of under-stemming. The words *lapsi* (a child), *oikeus* (a right), *yhdistynyt* (united) and *julistus* (convention) all occur in their inflected forms: *lasten* (children's), *oikeudet* (rights), *yhdistyneiden* (of united) and *julistuksesta* (of the convention). These are stemmed to following strings, respectively: *last*, *oikeud*, *yhdistyn* and *julistuks*, while the stems of the basic forms are: *lap*, *oikeus*, *yhdistyny* and *julistus*. These under-stemmings cause loss of recall. See Examples 15 and 16 in the Appendix.

*Unrecognized words*

A stemmer gives a stemmed string for all the input words, regardless of the origin of the word. It treats foreign words similarly as words belonging to the language. This is naturally an advantage from IR point of view. A lemmatizer is able to give the basic form only for words which it recognizes. For unrecognized words, for example many foreign names, some other techniques have to be applied. The simplest approach is to leave the word as such. We applied this approach in our monolingual runs.

*Finnish monolingual*. The Finnish topic 185 includes the word *Srebrenica* in inflected forms*: Srebrenicasta* (from Srebrenica) and *Srebrenicassa* (in Srebrenica). These strings do not match with index word *Srebrenica* (Srebrenica), which occurs in many relevant documents. In the stemmed index and stemmed run, the word *Srebrenica* is stemmed to a string *srebrenic* in all cases. See Examples 17 and 18 in the Appendix.

*Swedish monolingual*. The Swedish topic 197 including the words *Dayton* and *Bosnien-Hercegovina* is a good example of the ability of the stemmer to handle all words analogously (whether they are foreign or customary language words). Examples 11 and 12 in the Appendix.

## 7. Discussion

Our first research question was: Does monolingual retrieval with normalization give significantly better results than retrieval without normalization? According to our results, the answer depends on the language: in English retrieval, no significant differences could be found, while non-English retrieval with normalization outperformed retrieval without normalization.

The second research question we raised was about performance of monolingual retrieval with stemming compared with retrieval in the lemmatized index with decompounding and without decompounding. No remarkable differences between the performances of these methods could be found. In all the monolingual runs, retrieval with stemming gave even a little better results than retrieval with lemmatization in the compound index. The greatest difference could be found in the monolingual Swedish runs: the run in the lemmatized compound index gave 19.1 % worse result, and the run in the stemmed index 13.7 % worse result, than the run in the lemmatized decompounded index. In the Finnish monolingual runs, the performance of the run with a lemmatized decompounded index was only 2.0 % better than that of a stemmed index. There seem to be many topics, where retrieval in a lemmatized decompounded index performed better than in a stemmed index, but also some opposite ones. All the queries which got better result in the stemmed index than in the lemmatized indexes included unrecognized words (for the lemmatizer). The stemmer treats all the words analogously, independently of whether the word belongs to the language vocabulary, while the lemmatizer we used was not able to handle unrecognized words. N-gram techniques might make the retrieval result of lemmatized indexes better, but presumably they could not reach as good results as stemmers. This suggests that at least some simple stemming techniques should be applied to unrecognized words both in indexing and retrieval stages prior to n-gram matching.

The third research question was which gives the best result in the bilingual runs: retrieval in the stemmed index, lemmatized decompounded index or lemmatized compound index. In our bilingual runs, retrieval in the indexes without decompounding gave inferior results compared to retrieval in the decompounded index. We found that the greatest performance differences in the bilingual runs occurred in the cases where the topic included phrases. The source language of the test runs was English, which is a phrase rich language. In Finnish, German and Swedish, compounds are used instead of phrases in most cases. Phrases are treated analogously in bilingual stemmed and lemmatized runs: first the parts of the phrase are normalized utilizing the English lemmatizer, then translated, and finally either a stemmer or a lemmatizer is applied to normalize the parts of the phrase. The performance differences are due to indexing: if decompounding is applied, the performance is better. So the parts of the phrases match the index in the case where a phrase is used in the source language and a compound in the target language. The two indexes without decompounding performed almost equally in all the bilingual runs.

The phrase / compound problem is a typical problem of bilingual runs, and should not be present in the monolingual runs. However, even in these runs, retrieval in the decompounded index gave better results in some queries compared with the indexes without decompounding. Decompounding seems to affect a kind of query expansion. Presumably, this feature could in some cases add noise in retrieval as well.

Under-stemming and over-stemming which are possible sources of bad performance in stemmed runs could be found mainly in the English – Finnish run and monolingual Finnish run. The other bilingual or monolingual runs did not include clear cases like that. This may be seen as a contribution of the large number of highly inflected words in Finnish, as well as the quality of the stemmers applied.

English was the source language of the bilingual runs in this study, which may have an impact on the results. If both the source and the target languages are compound languages or the source language is a compound language and the target language is a phrase language, the compound problems might be less frequent. Another interesting research problem is whether the results could be improved utilizing English phrase recognition and a more extensive translation dictionary including English phrases.

## 8. Conclusions

Earlier research in IR shows that language has its impact on the performance of normalization in monolingual retrieval. With highly inflectional languages, normalization is capable to improve the retrieval result. With monolingual English retrieval, stemming has only minor impact on the retrieval result. (Harman 1991, Popovic & Willet 1992, Lennon & al, 1981, Alkula 2000, Braschler & Ripplinger 2004, Hollink & al. 2004). The results of the current study are in line with earlier research: normalization tools do not remarkably improve the retrieval result of monolingual English runs, but in non-English runs they give significantly better results.

Lemmatizers are useful for normalizing highly inflected languages. They have not been widely tested in IR research, probably because of the dominance of English, and their high prices. This research shows that retrieval in the index without decompounding utilizing stemmers performs as well as retrieval using lemmatizers in monolingual and bilingual IR, even with highly inflected languages. It is useful to know that there is inevitably no need to use lemmatizers, which are often commercial products with high licence fees.

The present study shows that retrieval in a decompounded index performs significantly better than retrieval in an index without decompounding in bilingual IR, when phrases are used in the source language, and compounds in the target language. Thus, to achieve better results, it is rational to use decompounded indexing in such bilingual tasks. In monolingual IR, the impact of decompounding on the retrieval result is not so remarkable.

Possible further research problems would include the following: does retrieval in a decompounded index outperform retrieval in an compound index, when 1) both the source and the target language are compound languages, or 2) the source language is a compound language and the target language is a phrase language.

## 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.

The SNOWBALL stemmers by Martin Porter.

## Appendix

*Example 1.*
English – Finnish query no. 187 with the lemmatized index
Average precision with decompounding 100 %
Average precision without decompounding 54.4 %

#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. 186 with the lemmatized index

Average precision with decompounding 91.0 %

Average precision without decompounding 45.3 %

#sum( #syn(holländsk) #syn( koalition) #syn( regering styrelsesätt styrande) #syn( politisk) #syn( fest sällskap parti party) #syn( ta form formera sig godkondition form formad pudding utkristallisera kast gestalt bänk figur format formulär tillstånd klass bilda utgöra) #syn( regera över styre regel tumstock tumstock fastställa) #syn( koalition) #syn( kalla på kalla kontakta ringa bjuda rop sång besök telefonsamtal lockrop lockrop efterfrågan skäl) #syn( purpur) #syn( skåp kabinett praktik) #syn(nederland holland nederländerna) #syn( 1994) #syn( 1995) )

*Example 4.*

English – Swedish query no. 186 with the stemmed index

Average precision 21.5 %

#sum( #syn(holländsk ) #syn( koalition) #syn( regering styrelsesät styr) #syn( politisk) #syn( fest sällskap parti party) #syn( god kondition form pudding utkristalliser kast gestalt bänk figur form formulär tillstånd klass bild ta form formera s utgör) #syn( styr regel tumstock regera över fastställ) #syn( koalition) #syn( kall kalla på kontak ring bjud rop sång besök telefonsamtal lockrop efterfråganskäl) #syn( purpur) #syn( skåp kabinet praktik) #syn (nederland holland nederländ )#syn( 1994) #syn( 1995) )

*Example 5.*

English – German query no. 184 with the lemmatized index

Average precision with decompounding 67.5 %

Average precision without decompounding 47.1 %

#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 lemmatized index

Average precision with decompounding 50.0 %

Average precision without decompounding 66.7 %

#sum( #syn(aasialainen) #syn( dinosaurus) #syn( maalliset jäännökset tähteet) #syn( jäädä edelleen jäädä) #syn( ranta lohko puolue osuus rannikko hiekkaranta äyräs rooli erota) #syn( asia tehtävä) #syn( dinosaurus) #syn( maalliset jäännökset tähteet) #syn( jäädä edelleen jäädä) #syn( pitää jonakin perustaa perustua löytää huomata) #syn( pitää jonakin löytää huomata löytö) )

*Example 8.*

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 9.*

English – Finnish query no. 174 with the lemmatized index

Average precision with decompounding 70.2 %

Average precision without decompounding 68.8 %

#sum( #syn( bavarian baijerilainen) #syn( krusifiksi) #syn( riita riidellä) #syn( pitää jonakin löytää huomata löytö) #syn( todistus huhu pamaus ilmoittaa ilmoittautua) #syn( krusifiksi) #syn( riita riidellä) #syn( bavarian baijerilainen) #syn(parvi koulu osasto yliopisto koulukunta koulia) )

*Example 10.*

English – Finnish query no. 174 with the stemmed index
Average precision 8.3 %
#sum( #syn( bavaria baijerilain) #syn( krusif) #syn( riita riide) #syn( löytä huoma pitää j löytö) #syn( todistus huhu pamaus ilmoit ilmoittautu) #syn( krusif) #syn( riita riide) #syn( bavaria baijerilain) #syn(parv koulu osasto yliopisto koulukun koul) )

*Example 11*.
Monolingual Finnish query no. 147 with the lemmatized index
Average precision with decompounding 41.8 %
Average precision without decompounding 2.0 %
#sum( #syn( öljyonnettomuus) #syn( lintu) #syn( etsiä) #syn( kertoa) #syn( tapaturmainen) #syn( öljyvuoto) #syn( öljysaaste) #syn( lintu) #syn(aiheuttaa) #syn( haitta) #syn( vamma) )

*Example 12*.
Monolingual Finnish query no. 147 with the stemmed index
Average precision 16.8 %
#sum( #syn( öljyonnettomuud) #syn( linu) #syn( et) #syn( dokument) #syn( jotk) #syn( kertov) #syn( tapaturmaist) #syn(öljyvuoto) #syn( öljysaast) #syn( linu) #syn( aiheuttam) #syn( haito) #syn( vamo) )

*Example 13*.
Monolingual Swedish query no. 197 with the lemmatized index
Average precision with decompounding 59.4 %
Average precision without decompounding 0.2 %
#sum( #syn( fredsavtal) #syn( dayton @dayton) #syn( leta) #syn( efter) #syn( rapport) #syn( null) #syn( fredsavtal)
#syn( från) #syn( dayton @dayton) #syn( föra) #syn( fred) #syn( bevara) #syn( null) #syn( bosnien @bosnien)
#syn( hercegovina @hercegovina) )

*Example 14*.
Monolingual Swedish query no. 197 with the stemmed index
Average precision 60.1 %
#sum( #syn( fredsavtal) #syn( dayton) #syn( let) #syn( eft) #syn( rapport) #syn( null) #syn( fredsavtalet) #syn( från)
#syn( dayton) #syn( för) #syn( fred) #syn( bevar) #syn( null) #syn( bosni) #syn( hercegovin) )

*Example 15*.
Monolingual Finnish query no. 152 with the lemmatized index
Average precision with decompounding 76.5
Average precision without decompounding 75.6
#sum( #syn( lapsi) #syn( oikeus) #syn( etsiä) #syn( tieto) #syn( yhdistyä) #syn( kansakunta) #syn( lapsi) #syn( oikeus)
#syn( julistus) )

*Example 16*.
Monolingual Finnish query no. 152 with the stemmed index
Average precision 13.5 %
#sum( #syn( last) #syn( oikeud) #syn( et) #syn( tieto) #syn( yhdistyn) #syn( kansakunt) #syn( last) #syn( oikeuks)
#syn( julistuks) )

*Example 17*.
Monolingual Finnish query no. 185 with the lemmatized index
Average precision with decompounding 50.0 %
Average precision without decompounding 50.0 %
#sum( #syn( hollantilainen) #syn( valokuva) #syn( srebrenicasta @srebrenicasta) #syn( tapahtua) #syn( valokuva) #syn( filmi) #syn( hollantilainen) #syn( sotilas) #syn( ottaa) #syn( srebrenicassa @srebrenicassa) #syn( tarjota) #syn( todiste) #syn( ihminenoikeus) #syn( loukkaus) )

*Example 18*.
Monolingual Finnish query no. 185 with the stemmed index
Average precision 100.0 %
#sum( #syn( hollantilaist) #syn( valokuv) #syn( srebrenic) #syn( mitä) #syn( tapahtui) #syn( niil) #syn( valokuv) #syn( film) #syn( joita) #syn( hollantilais) #syn( sotil) #syn( ottiv) #syn( srebrenic) #syn( jotk) #syn( tarjosiv) #syn( todist) #syn(ihmisoikeuks) #syn( loukkauks) )

## References

Airio E, Keskustalo H, Hedlund, T and Pirkola A (2003) UTACLIR @ CLEF2002 – Bilingual and multilingual runs with a unified process. In: Peters C, Braschler M, Gonzalo J and Kluck M, Eds., Advances in cross-language information retrieval. Results of the cross-language evaluation forum - CLEF 2002. Lecture Notes in Computer Science 2785. Springer, pp. 91-100.

Alkula R (2000) Merkkijonoista suomen kielen sanoiksi. Ph D. Thesis, University of Tampere, Department of Information Studies, Acta Universitatis Tampererensis 763. Acta Electronica Universitatis Tamperensis 51. http://acta.uta.fi/pdf/951-44-4886-3.pdf.

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

Harman, D (1991) How effective is suffixing? Journal of the American Society for Information Science, 42(1):7-15.

Hedlund T, Pirkola A and Järvelin K (2001) Aspects of Swedish morphology and semantics from the perspective of mono- and cross-language information retrieval. Information Processing and Retrieval, 37(1):147-161.

Hedlund T, Keskustalo H, Pirkola A, Airio E and Järvelin K (2002a) UTACLIR @ CLEF 2001 – Effects of compound splitting and n-gram techniques. In: Peters C, Braschler M, Gonzalo J and Kluck M, Eds., Evaluation of cross-language information retrieval systems. Second workshop of the cross-language evaluation forum, CLEF 2001. Lecture Notes in Computer Science 2406. Springer, pp. 118-136.

Hedlund T, Keskustalo H, Airio E and Pirkola A (2002b) UTACLIR : An extendable query translation system. In: Gey FC, Kando N and Peters C, Eds., SIGIR 2002 Workshop I, Cross-language information retrieval: a research map. University of Tampere, Finland, August 15, 2002.

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

Hull, DA (1996) Stemming algorithms: A case study for detailed evaluation. Journal of the American Society for Information Science, 47(1): 70-84.

Kettunen K, Kunttu T and Järvelin K (2004) To stem or lemmatize a highly inflectional language in a probabilistic IR environment? Journal of Documentation, 61(xxx): xxx-xxx, accepted with minor revision.

Koskenniemi K (1983) Two-level morphology: A general computational model for word-form recognition and production. University of Helsinki, Finland. Publications No. 11.

Koskenniemi K (1985) A general two-level computational model for word-form recognition and production. In: Karlsson F, Ed., Computational morphosyntax. Report on research 1981-84. Publications No. 13. University of Helsinki, Department of General Linguistics, pp. 1-18.

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

Kraaij W (2004) Variations on language modeling for information retrieval. CTIT PhD. –thesis No. 04-62, University of Twente.

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 Retrieval. Seattle, WA, pp 191 - 202.

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

Lennon M, Peirce DS, Tarry BD and Willet P (1981) An evaluation of some conflation algorithms for information retrieval. Journal of Information Science, 3(1981): 177-183.

McNamee P, Mayfield J (2001) A language-independent approach to European text retrieval. In Peters C, Ed, Cross-Language Information Retrieval and Evaluation: Proceedings of the CLEF-2000 Workshop, Lecture Notes in Computer Science 2069, Springer, Lisbon, Portugal, pp. 129-139.

Niedermair GT, Thurmair G and Büttel I (1984) MARS: a retrieval tool on the basis of morphological analysis. In: van Rijsbergen CJ, Ed., Research and development in information retrieval. Cambridge University Press, pp. 369-381.

Pirkola A (1998) The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA, pp 55-63.

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

Popovic M and 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 (1980) An algorithm for suffix stripping. Program, 14(3):130-137. http://telemat.det.unifi.it/book/2001/wchange/download/stem_porter.html.

Porter M (1981) Snowball: A language for stemming algorithms. http://snowball.tartarus.org/texts/introduction.html (visited January 7th, 2004).