Information Retrieval, 7, 291–316, 2004

How Effective is Stemming and Decompounding for German Text Retrieval?

MARTIN BRASCHLER martin.braschler@eurospider.com; martin.braschler@unine.ch
Eurospider Information Technology AG, Schaffhauserstrasse 18, CH-8006 Zürich, Switzerland; Université de Neuchâtel, Institut Interfacultaire d'Informatique, Pierre-à-Mazel 7, CH-2001 Neuchâtel, Switzerland

BÄRBEL RIPPLINGER BRipplinger@web.de
Eurospider Information Technology AG, Schaffhauserstrasse 18, CH-8006 Zürich, Switzerland

Received January 20, 2003; Revised May 14, 2003; Accepted September 10, 2003

Abstract. Information retrieval systems operating on free text face difficulties when word forms used in the query and documents do not match. The usual solution is the use of a "stemming component" that reduces related word forms to a common stem. Extensive studies of such components exist for English, but considerably less is known for other languages. Previously, it has been claimed that stemming is essential for highly declensional languages. We report on our experiments on stemming for German, where an additional issue is the handling of compounds, which are formed by concatenating several words. The major contribution of our work that goes beyond its focus on German lies in the investigation of a complete spectrum of approaches, ranging from language-independent to elaborate linguistic methods. The main findings are that stemming is beneficial even when using a simple approach, and that carefully designed decompounding, the splitting of compound words, remarkably boosts performance. All findings are based on a thorough analysis using a large reliable test collection.

Keywords: stemming, decompounding, German, evaluation, morphological analysis

1. Introduction

Most modern information retrieval (IR) systems implement some sort of what is commonly known as "stemming". A system using stemming conflates derived word forms to a common stem. The benefits of such a procedure is two-fold: by conflating several forms to the same stem, the number of entries and the size of the search index is reduced. More importantly, stemming is potentially helpful for free text retrieval, where search terms can occur in various different forms in the document collection. Stemming makes retrieval of such documents independent from the specific word form used in the query. In our experiments, we will concentrate on this second aspect, the impact of stemming on retrieval effectiveness.

The main reason for the use of stemming is the hope that through the increased number of matches between search terms and documents, the quality of search results is improved. In terms of the most popular measures for determining retrieval effectiveness, precision (amount of relevant documents retrieved compared to all documents retrieved) and recall (amount of relevant documents retrieved compared to total number of relevant documents in the collection), stemmed terms retrieve additional relevant documents that would have otherwise gone undetected, i.e. they improve recall. There is also potential for improved precision,

since additional term matches can contribute to a better weighting for a query/document pair.

Viewing stemming as a vehicle to enhance retrieval effectiveness necessitates an analysis of the produced stems with regard to overstemming or understemming. If a stemmer conflates terms too aggressively, thereby conflating unrelated or loosely related words, many extraneous matches between the query and irrelevant documents are produced; we talk of "overstemming". Even though some of these stems may be correct from a linguistic viewpoint, they are not helpful for retrieval. In contrast, if crucial relevant documents are missed because of a conservative stemming strategy, we speak of "understemming". A good stemmer has to find the right balance of conflation for effective retrieval.

Where many previous studies compare their stemming method only to no stemming or to simple affix stripping methods such as proposed by Lovins (1968) and Porter (1980), the experiments described here compare a complete spectrum of methods, ranging from language-independent to sophisticated linguistic analysis. We consider this an important step forward, since while smaller studies may demonstrate the benefit of stemming for a certain language in principle, they cannot answer the important question of how to build a component that maximizes the potential of stemming for that language. Only an exhaustive comparison can give an indication of the right amount of linguistic knowledge and the right balance of conflation that is necessary.

The paper deals with the German language, where, besides stemming, decompounding seems to be an additional issue in retrieval. In most Germanic languages (e.g. German, Dutch, Swedish), but also in some other languages (e.g. Finnish), it is possible to build compounds by concatenating several words. Performance, especially recall, may be negatively affected if this word formation process is not taken into account. If the user enters only parts of compound words or replaces the compound with a phrasal construction, relevant documents may not be found.

Similar to stemming, there are different approaches to analyze and split compounds ("decompounding"). We concentrate on the usefulness of compound splitting for information retrieval, and not as a linguistic exercise. The splitting of some compounds words can cause a shift of meaning. Such compounds should be left intact ("conservative decompounding"). On the other hand, the productive nature of German compound formation, which allows for ad-hoc building of new compounds, makes it impossible to enumerate all potential compound words. "Aggressive decompounding" attempts to produce a maximum number of splittings by using heuristics. In our experiments, we investigated different approaches to decompounding, ranging from surface-based to well-developed linguistic methods.

As a consequence of limited resources for some languages, many studies on stemmer and decompounding behavior have been conducted using small test collections, containing short documents. Both the size of the collection and the length of the documents can influence observations on stemming. Especially the length of the retrieval items is important, since short documents (e.g. only titles, or titles and abstracts) increase the likelihood for word mismatch if no stemming is used. It is therefore not immediately obvious if a performance improvement measured on collections with short documents, such as widely used before 1990, translates into an improvement on larger collections with long documents. Furthermore, large test collections may be needed to draw more reliable conclusions. In our work

we used a part of the large German collection from the CLEF corpus to get appropriate statistical data. The collection contains lengthy newspaper and news magazine articles.

Similarly, the characteristics of the queries can impact the conclusions of the stemming experiments. Again, longer queries will have more likelihood of having multiple forms of search terms included, which may affect the amount of new information that a stemming component can contribute. As a consequence, we have used two different query lengths in all our experiments.

The remainder of the paper is structured as follows: we will first give an overview of related work (Section 2) and a summary of some relevant key characteristics of the German language (Section 3), before detailing the experimental setup in Section 4 (test collection), Section 5 (stemming and decompounding approaches) and Section 6 (retrieval system). The experiments themselves are then discussed, along with a careful statistical analysis and a query-by-query analysis (Section 7 through Section 9). The paper closes with conclusions and an outlook.

2. Related work

The role of stemming is well explored for English, even though results are controversial. Where Harman (1991) reported that stemming gives no benefit, Frakes (1992) and Hull (1996) claim at least a small benefit. Especially notable is the study by Hull (1996), which describes a very careful analysis of different stemmers and provides an ambitious attempt to compare a wide range of different approaches to stemming. Furthermore, in this study, data from the popular TREC test collections (Harman 1997) is used, which are much larger than earlier collections and contain lengthy documents. Because English has no productive compounding process, we are not aware of studies into the effects of decompounding for this language.

In recent years, more studies on stemming behavior for languages other than English have been published. While simple stemmers are considered sufficient for languages such as English, it is claimed that rich declensional languages require more sophisticated stemming approaches. Studies exist on Slovene (Popovic and Willet 1992), Hebrew (Choueka 1992), Dutch (Kraaij and Pohlmann 1996b), German (Womser-Hacker 1989, Moulinier et al. 2001), Italian (Sheridan and Ballerini 1996), French (Savoy 1999), and others. For most of these languages, significant benefits (from 10% to 130%) can be observed. Decompounding has been investigated, at least for German (Moulinier et al. 2001) and Dutch (Kraaij and Pohlmann 1996a). The improvements reported are about 17% to 54%.

Previous work on what could be termed as "German stemming" was mainly conducted in the context of studies comparing manual and automatic indexing in databases. The PADOK studies (Womser-Hacker 1989, Krause and Womser-Hacker 1990), carried out in the late eighties, compared several approaches to automatic indexing on patent data. The baseline system indexed every token in the original form as it appears in the document. However, queries could make use of string truncation and wildcards. This baseline system was compared to the two systems "PASSAT", comprising morphological analysis, including decompounding (used in PADOK I & PADOK II), and "CTX" which additionally applied a syntactical analysis (used in PADOK I only). In PADOK I (Womser-Hacker 1989),

documents consisting of titles and abstracts only were used. It was found that PASSAT achieved higher recall and CTX achieved higher precision compared to the baseline system. In PADOK II (Krause and Womser-Hacker 1990), the full text of the documents was used. In this new setting, no significant gain from using the morphological analysis employed by PASSAT could be detected.

The later GIRT pretest used texts from the field of social sciences (Frisch and Kluck 1997). Two systems were compared: "freeWAISsf" and "Messenger", differing in more areas than just in the stemming approaches employed (freeWAISsf uses weak morphological analysis, Messenger apparently no stemming). This makes it hard to draw conclusions on the effectiveness of the stemming component based on the published results.

Recent work by Moulinier et al. (2001) reported a performance gain of 20% using linguistic analysis (stemming plus compound analysis) for German. A gain of 89% in recall together with a loss of 24% in precision are the results obtained by Ripplinger (2002) in a small experiment within a domain specific environment. The experiments by Monz and de Rijke (2002) show a gain of 25% to 69% for German and Dutch respectively using blind relevance feedback in addition to a linguistic processing. Tomlinson (2002) applied linguistic stemming including decompounding, and obtained a performance improvement of 43% for German, and 30% for Dutch. He also conducted experiments for English and Romance languages (French, Italian, and Spanish) with a performance gain of 5% to 18%.

Most of this recent work came in the context of the TREC CLIR task and the CLEF campaigns. The experiments are usually confined to a comparison between one particular stemming or decompounding approach and not using any form of stemming at all. It is therefore unclear how the findings can assist in the choice of a particular stemming approach or how to assess the absolute potential of stemming for performance improvement in the German language.

3. Characteristics of the German language

German, in contrast to English, is a highly declensional language, a fact expressed by a rich system of inflections and cases. Depending on the word class, i.e. noun, verb, or adjective, there is a set of possible inflections for each particular word (e.g. 144 forms for verbs). For instance, "informiert", "informiere", "informierte", "informierten" are forms of the verb "informieren" (to inform). Furthermore, words can be formed by attaching multiple derivational inflections to a stem in order to build new forms. For instance, the lexeme "inform" is the stem for "informieren", "informierend", "Information", "Informant", "informative", "informatorisch", but not "informal". To ensure efficient retrieval all these forms should be conflated to the same stem.

Additionally, in most Germanic languages (e.g. German, Dutch, Swedish), as well as in some other languages (e.g. Finnish), compounds can be built by concatenating several words, such as e.g., "Haus|tür" (house door), "Wind|energie" (wind energy), "Früh|stück" (breakfast) or "Unglücks|fall" (accident). Such compound formation occurs in almost all languages, such as "hair|dresser", "speed|boat" in English, and "porte|feuille" (wallet), "bon|homme" (fellow) in French. In English and in Romance languages, however, these words are lexicalized, i.e. they cannot be expressed by a nominal phrase the way

compounds in Germanic languages could be ("Haustür" vs. "Tür des Hauses"). Because Germanic languages also know lexicalized compounds (e.g. "Wettbewerb", "Frühstück"), the correct treatment of such words is quite complicated. Therefore, we expect that compound analysis requires more linguistic knowledge to be of benefit than stemming does.

In principle, only words with certain types of part-of-speech can be coupled, for instance noun/noun, adjective/noun, or verb/noun. Some analyzers split only those compounds where the constituents have the same part-of-speech (noun/noun, adjective/adjective), and could be thus classified as "conservative". Others split the compounds into all possible word forms (often by means of a lexicon lookup). Because these methods do not consider the part of speech of the constituents, i.e. also splitting into pronouns, prepositions or articles, this approach can be described as "aggressive". Linguistic compound analyzers lie in between, splitting compounds only into valid word forms, i.e. nouns, verbs, adjectives.

4. The test collection

Many of the studies on stemming behavior for languages other than English suffer from scarce availability of suitable test collections. Whereas for English the well-known TREC collections exist, comparable test data became available only recently for important European and Asian languages (through the CLEF and NTCIR campaigns). For their 2001 evaluation campaign, the CLEF consortium distributed a multilingual document collection of roughly 1,000,000 documents written in one of six languages (Dutch, English, French, German, Italian and Spanish). For the German part of this data, CLEF 2000 used articles from the daily newspaper "Frankfurter Rundschau" and the weekly news magazine "Der Spiegel". CLEF 2001 used a superset of this, adding newswire articles taken from the "Schweizerische Depeschenagentur SDA". To go with this data, there is a total of 90 topics[1] (40 for 2000, 50 for 2001) with corresponding relevance assessments. By eliminating the SDA articles, we were able to form a unified German test collection, using the topics from both years. Of the 90 topics, 85 have matches in the German data that we used. This comparatively high number of topics (many studies use only 50 queries or less) facilitates the detection of significant differences in stemming and decompounding behavior (see also Table 1 for more details of the test collection).

5. Stemming and decompounding approaches

One of the main objectives of this study is to compare approaches from a spectrum as wide as possible. This spectrum covers methods ranging from completely language-independent methods to components that use elaborate linguistic knowledge. In our experiments, apart from using different stemming and decompounding methods, all other indexing parameters remain constant (tokenization, stopword list, etc.).

To keep things readable, the names of the different approaches are abbreviated in some places using a letter-based abbreviation scheme. The respective "codes" are given in the following subheadings. Approaches applying stemming only are denoted by lower case

Table 1. Key characteristics of the test collection.

Document source	Frankfurter Rundschau 1994, Der Spiegel 1994, 1995
Number of documents	153,694
Size (MByte)	383
Number of topics	85[a]
Number of indexing tokens per document (tokens minus stopwords)	
Maximum	2,885
Minimum	1
Mean	156.09
Median	128
Relevance assessments:	
Number of assessments:	20,980
Number of relevant documents	1,790
Maximum relevant (one topic)	109
Minimum relevant (one topic)	1
Mean relevant (one topic)	21.06
Median relevant (one topic)	12

[a]90 from CLEF multilingual topic sets, minus 5 topics without any relevant documents in the subset of the German data we used.

letter codes, while those deploying decompounding are marked by capitalizing the initial letter.

The approaches used in this study are:

5.1. No stemming ("n" "nostem" run)

As a baseline, we indexed all documents without using any form of stemming or decompounding.

5.2. Combination of word-based and n-gram based retrieval ("6" "6gram + word" run)

The use of combined character *n*-gram and word-based indexing was reported as a successful approach to German text retrieval by Mayfield et al. (1999) and Savoy (2003). Based on their findings, we chose to combine 6-grams and unstemmed words. The individual 6-grams, built on the unstemmed words, potentially span word boundaries. A main benefit of this approach is its complete language independence—no specific linguistic knowledge is needed to form the *n*-grams, and the word-based indexing is done without attempting conflation. On the other hand, the method is storage-intensive: the large number of different *n*-grams leads to a massively increased index size (roughly three times that of an unstemmed word-based index).

5.3. Linguistica: Automatic machine learning ("l" "limguistica" run)

Linguistica (Goldsmith 2001) performs a morphological segmentation based on unsupervised learning. The aim is to find the correct morphological splits for individual words, in a language-independent way. The program first establishes for all words a "best splitting" into two parts of two or more letters (not to be confused with splitting compounds). These are then treated as a "stem" and a "suffix", even though they may in reality represent other units, such as a prefix and a stem. If multiple candidates for such a splitting exist, heuristics based on word frequency and the number of possible parses are applied. In a second step, the program determines a list of suffixes that can occur with each stem, called the "signature" of a stem. Each stem is associated with exactly one signature. For example, in our experiments, the stems "erzeug", "verpack", "befrag" and "beschaff" were associated with the signature "en.er.t.te.ung", meaning that Linguistica learnt splittings such as "verpack-en" and "befrag-ung" from the test collection.

As a result, Linguistica outputs a lists of signatures, possible affixes, and the list of words contained in the processed corpus together with their possible splits (the "lexicon"). For the test collections (consisting of roughly 1.4 million unique terms) the lexicon contains about 123.000 entries, for instance "machbar" (feasible), "machbar-en" (feasible), "machbar-es" (feasible), "machbar-keit" (feasibility). For our experiment we used these entries, which often do not really denote a proper stem from a linguistic viewpoint.

Decompounding is not a feature in Linguistica, however, the program is sensitive to words used frequently in compounds such as "bank" (bank), "partei" (party) and "teil" (part). These are categorized as affixes, leading to signatures such as:

NULL.bank.bild.dienste.haus.partei.plan.rat.staat.teil.viertel
(*NULL.bank.picture.service.house.party.plan.council.state.part.quarter*)

All values are valid lexemes (i.e. nouns) in German. Such signatures result in accidental decompounding: since suffixes are discarded after stemming, compounds such as "Datenbank" (data base), "Anklagebank" (dock), "Handelsbank" (merchant bank) are conflated to "Daten" (data), "Anklage" (charge) and "Handel" (trade). The second component of the compound noun is lost and cannot be used for subsequent retrieval.

5.4. NIST stemmer: Rule-based approach ("t" "nist_stem" run, "T" "nist_dec" run)

The NIST German rule-based stemmer has been constructed through analysis of the frequency of German suffixes in large wordlists. Its approach is similar to the Porter English stemmer (Porter 1980). The rules were hand-crafted to produce as many valid conflations of high-frequency word forms as possible, while keeping the rate of incorrect conflations low. Rules operate based on the suffix to be replaced, the new suffix used as the replacement, the length of the word, and additional, special criteria. The length of the word is calculated using a special measure introduced by Porter that counts the number of consonant/vowel sequences (informally connected to the number of syllables a word contains). Additional criteria include among others whether the word contains at least one vowel, or if the word

ends in special character combinations. The stemmer attempts to iteratively strip suffixes from a word, applying the rules cyclically. The resulting stems are often not correct from a linguistic viewpoint. For instance the word "glück-lich-er-weis-e" (luckily) is reduced to "gluck" (same stem as for "luck"). The stemmer is available as a part of the NIST ZPrise 2 retrieval system.

The stemmer can be combined with a corpus-based decompounding component based on co-occurrence analysis. After collecting a list of candidate nouns, the component tries to find valid splittings by looking for potential constituents that co-occur in the same documents. In case more than one potential splitting is identified, a heuristic is used for selection. Highest preference is given to splittings that result in a maximum number of constituents. This purely corpus-based approach produces a number of errors, but is overall rather conservative in the number of splittings generated.

For our experiments, we used both the pure stemming approach ("t" "nist_stem" run) and the combination with decompounding ("T" "nist_dec" run). This study represents the first careful evaluation of the NIST German stemmer.

5.5. *Spider stemmer: Commercially motivated (rule-based and lexicon) approach ("sn" "spider_NS", "Sf" "spider_FS", "Sc" "spider_CS", "Ss" "spider_SS" runs)*

For our experiments we had access to the stemming component used in the commercial "*relevancy*" retrieval system by Eurospider, the "Spider stemmer". The approach is based on a combination of a lexicon and a set of rules that are used for suffix stripping and unknown words (Wechsler et al. 1997).

The Spider stemmer has been used for over five years in all commercial installations of the system, and has therefore been constantly adapted according to customer feedback. Extensive performance figures for this stemmer are published publicly for the first time in this study.

The component includes optional decompounding, which can be applied in one of three modes, from conservative to aggressive splitting. For our runs we used the version without decompounding (run "sn" spider_NS) as well as three forms of decompounding: a conservative option (splits words only if all the components have the same part-of-speech as the overall compound, run "Ss" "spider_SS"), a more relaxed option (at least one component has to match the overall compound with regard to part-of-speech, run "Sc" "spider_CS") and an aggressive version (split whenever possible, run "Sf" "spider_FS").

For example, "Umweltaspekte" (environmental aspects) is reduced to "umweltaspekt" (run "sn") and split to "um-welt-aspekt" (run "Sf").

5.6. *MPRO: Morpho-syntactic analysis ("mu" "MPRO_lu" run, "ms" "MPRO_ls" run, "Mu" "MPRO_lu_dec" run, "Ms" "MPRO_ls_dec" run)*

MPRO, a development by the IAI (Maas 1996), performs a morpho-syntactic analysis consisting of lemmatization, part-of-speech tagging, and, for German, a compound analysis. To lemmatize and tag words with their part-of-speech(s), MPRO uses general morphological rules in form of small subroutines that co-operate with a morphological dictionary. As

a result, for each word a set of attribute-value pairs describing inflectional attributes (e.g. gender, number, tense, mood, etc.), word structure and semantics (lexical base form, derivational root form, compound constituents, semantic class, etc.) is produced. For instance, the analysis of the word "Kollisionen" (collisions) results in

{string = kollisionen, c = noun, lu = kollision, nb = pl, g = f, t = kollision,
ls = kollidieren, s = ation, . . .}

and produces the lexical unit "kollision" and the root form "kollidieren" (collide). For the compound "Umweltaspekte" (environmental aspects) MPRO generates a splitting based on lexical base forms, "umwelt-aspekt" (t feature), and one based on derivational root forms of the compound constituents, (ls feature; in this example identical).

{string = umweltaspekte, c = noun, lu = umweltaspekt, nb = pl, g = f,
t = umwelt_aspekt, ls = umwelt_aspekt, . . .}

With this tool, the corpus has been analyzed and for each word, information about the lexical base form (lu), and the derivational root (ls) is used to generate two lexical resources. The derivational root form presents a kind of "aggressive" stemming and splitting because the morphological derivation is more general than the lexical base form. For a more detailed description how to deploy MPRO for IR purposes see Ripplinger (2002).

We have used either the lexical base forms or the root forms and no decompounding information for the stemming only runs, "mu" "MPRO_lu" and "ms" "MPRO_ls", and the splitting based on lexical base forms or on derivational information for the decompounding runs, "Mu" "MRPO_lu_dec" and "Ms" "MPRO_ls_dec".

6. The retrieval system

For retrieval, the commercial "*relevancy*" system from Eurospider is used. Since the system uses a modular approach, the lexical resources generated from the individual approaches described above can easily be used for indexing. "*relevancy*" also offers the possibility of indexing *n*-grams as described above. For weighting of both documents and queries, we chose the Lnu.ltn weighting scheme (Singhal et al. 1996), with the "slope" parameter set to 0.15 (determined empirically to be the optimal value). Lnu.ltn is widely used and has given competitive performance on the CLEF collection (see e.g. Braschler and Schäuble 2001). The scheme includes document length normalization, a factor influenced by stemming, and some caution may be appropriate when generalizing the results to approaches that use vastly different collection statistics for weighting. There is no different weight for compound constituents as proposed by Hull et al. (1996).

7. Experiments

For each of the approaches described above, we produced several results sets ("runs"), using different sections of the CLEF topics. The topics are structured into title, description

and narrative fields, which express a user's information need in different length. The title section (T) typically consists of one to three keywords, the description section (D) typically consists of one sentence, and the narrative section (N) usually is several sentences long, giving detailed instructions on how to determine the relevance of retrieved documents. Using these fields, seven different combinations are possible (TDN, TD, TN, DN, N, D, T). In this paper, we report on the results obtained using the longest possible queries TDN and the shortest possible queries T, two popular choices for retrieval experiments.

To clearly present the results in the following tables and graphs the runs have been divided into several groups. In tables, runs of the two query lengths TDN and T are distinguished. Tables are also divided into approaches using decompounding (top half) and those using stemming only (lower half), separated by a dotted line. The baseline of no stemming is given at the bottom of the table. The runs have been structured further for the graphs. We give graphical results for both query lengths, one overview graph each for runs with and without decompounding. In these overview graphs, the representation of the Spider and MPRO stemmers is limited to one variant each. We have chosen the best performing options: "Sf" "spider_FS" for Spider/decompounding, "sn" "spider_NS" for Spider/stemming only, "Ms" "MPRO_ls_dec" for MPRO/decompounding, and "mu" "MPRO_lu" for MPRO/stemming only. A graphical presentation of the results of the other variants is given in separate graphs. In all graphs, "no stemming" is represented as the baseline for reference.

In Table 2, we give the mean average precision numbers for all runs. This is the most popular single-valued measure for TREC-style retrieval experiments, and is defined as non-interpolated average precision over all relevant documents. According to these numbers, methods that use decompounding (lower half of Table 2) perform better than methods that do not split compound words (upper half of Table 2). An exception is the combined

Table 2. Retrieval results (mean average precision). Runs using decompounding are given in the upper half, whereas the runs in the lower half use stemming only.

Run (T)	Avg prec (T)	Run (TDN)	Avg prec (TDN)
"Sf" spider_FS	0.3650 (+60.4%)	"Sf" spider_ FS	0.4471 (+34.6%)
"Ss" spider_ SS	0.3586 (+57.6%)	"Ms" MPRO_ls_dec	0.4440 (+33.7%)
"Ms" MRPO_ls_dec	0.3547 (+55.9%)	"Ss" spider_SS	0.4431 (+33.4%)
"Sc" spider_CS	0.3546 (+55.9%)	"Sc" spider_CS	0.4415 (+32.9%)
"Mu" MPRO_lu_dec	0.3385 (+48.8%)	"Mu" MPRO_lu_dec	0.4234 (+27.5%)
"T" nist_dec	0.3240 (+42.4%)	"T" nist_dec	0.4022 (+21.1%)
"6" 6gram + word	0.2757 (+21.2%)	"6" 6gram + word	0.3219 (−3.1%)
"t" nist_stem	0.2792 (+22.7%)	"t" nist_stem	0.3682 (+10.9%)
"sn" spider_NS	0.2722 (+19.6%)	"sn" spider_NS	0.3616 (+8.9%)
"mu" MPRO_lu	0.2682 (+17.9%)	"mu" MPRO_lu	0.3461 (+4.2%)
"ms" MPRO_ls	0.2353 (+3.4%)	"l" linguistica	0.3435 (+3.4%)
"l" linguistica	0.2302 (+1.2%)	"ms" MPRO_ls	0.3396 (+2.3%)
"n" nostem	0.2275	"n" nostem	0.3321

6-gram/word-based indexing run, which we treat as a "decompounding run", since the 6-grams allow matching of compound parts. This is the only run that for long queries (TDN) performs worse than the baseline of no stemming. The best variants of the two top methods, "Sf" "spider_FS" and "Ms" "MPRO_ls_dec", show almost identical performance, while "T" "nist_dec" performs slightly worse (see also figures 2 and 4). They obtain performance gains of between 33% and 60% depending on query length when compared to no stemming. The gap to the best method without decompounding ("t" "nist_stem") is fairly large—"Sf" "spider_FS" outperforms "t" "nist_stem" by 30.7% (T queries) and 21.4% (TDN queries), respectively. The individual variants of the Spider and MPRO stemmers show very little difference. For both query lengths, the most aggressive "Full Split" variant of the Spider stemmer slightly outperforms the other two decompounding variants (see also figures 5 and 6). For MPRO, the difference between the "Ms" "MPRO_ls_dec" and "Mu" "MPRO_lu_dec" decompounding variants is somewhat larger (see also figures 7 and 8). Again, the "Ms" "MPRO_ls_dec" variant using the root forms can be regarded as the more "aggressive" variant, and has the upper hand. The reverse, however, is true for the runs that use stemming only: the "mu" "MPRO_lu" run using the lexical units outperforms the "ms" "MRPO_ls" run for both query lengths.

The top three non-decompounding methods, "t" "nist_stem", "sn" "spider_NS", and "mu" "MPRO_lu", have similar performance for both query lengths, as does the 6-gram/word-based run, but for T queries only (see also figures 1 and 3). The latter performs clearly worse for TDN queries.

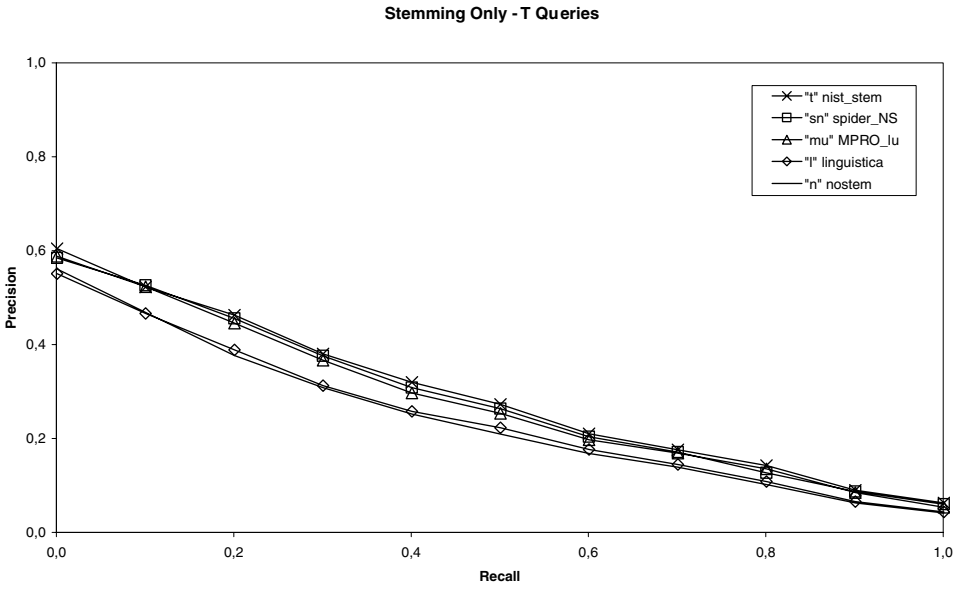

Figure 1. Recall/precision curve for "stemming only" runs (short T queries). The top three methods show very similar performance.

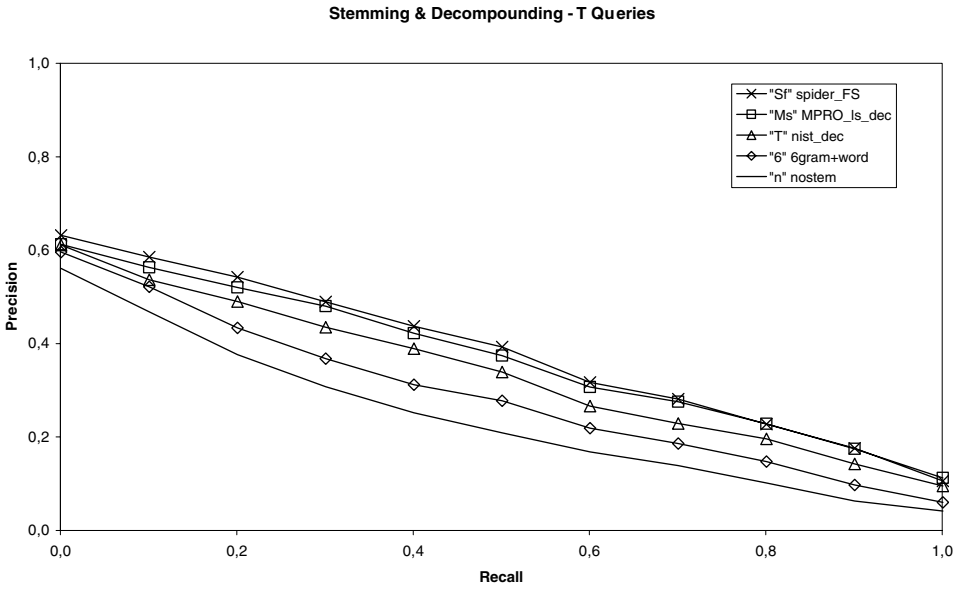

Figure 2. Recall/precision curve for runs using stemming and decompounding (short T queries). The choice of decompounding method produces much larger performance differences than those observed for "stemming only" experiments.

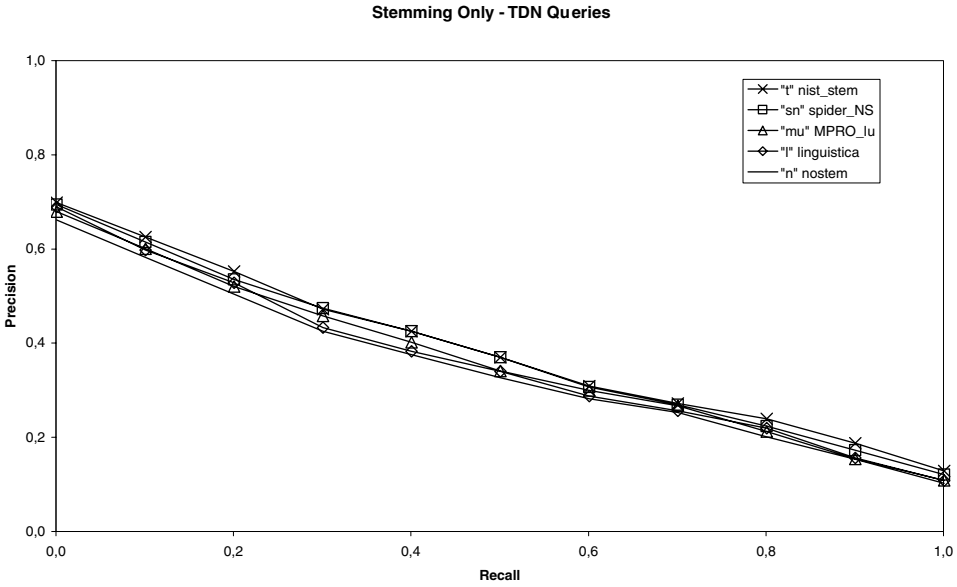

Figure 3. Recall/precision curve for "stemming only" runs (long TDN queries). As for short queries, the performance of the top methods is very similar.

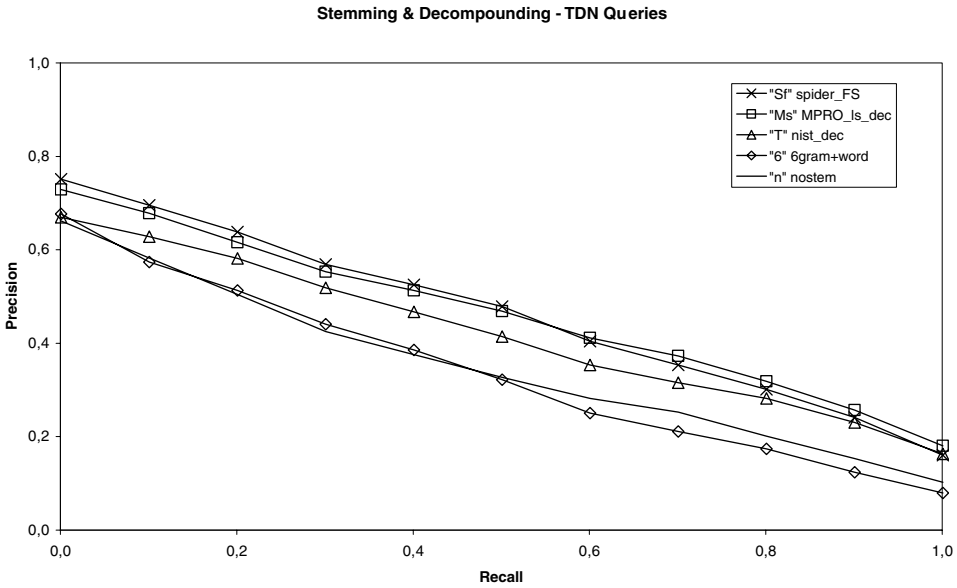

Figure 4. Recall/precision curve for runs using stemming and decompounding (long TDN queries). As for short queries, the performance differences are more evident than for the "stemming only" experiments.

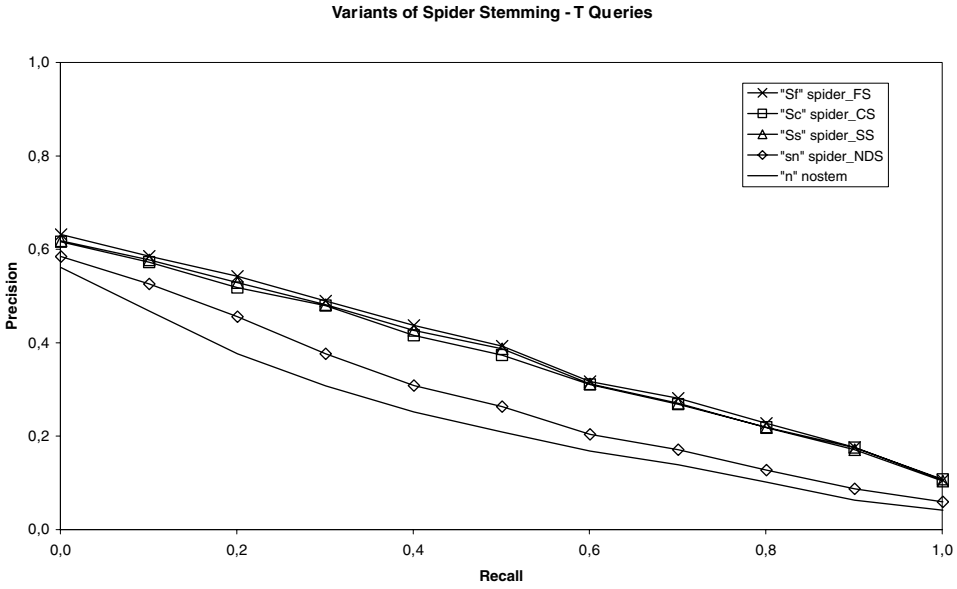

Figure 5. Variants of Spider stemming (short T queries). Given are the three decompounding variants, the stemming only variant, and the baseline (no stemming).

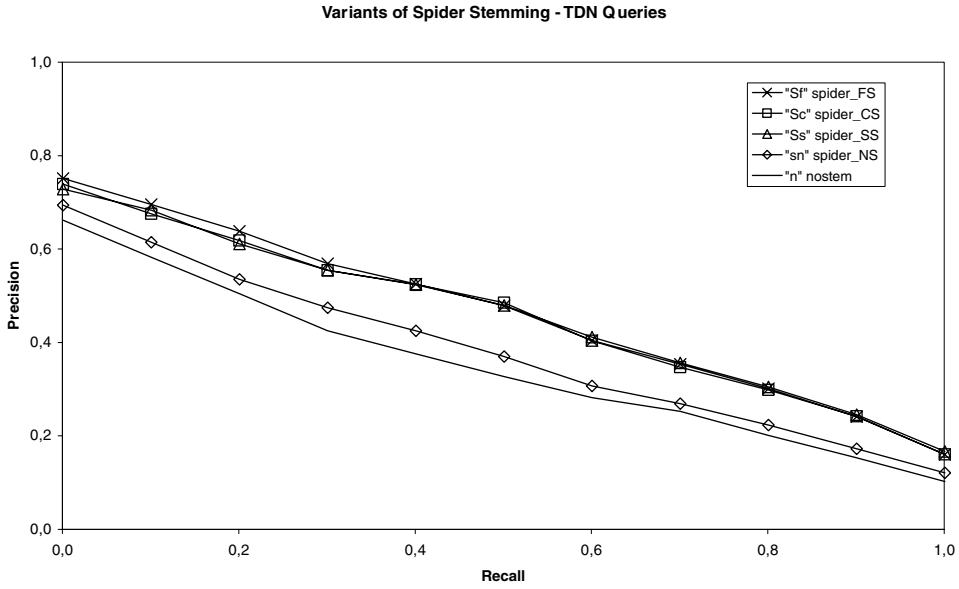

Figure 6. Variants of Spider stemming (long TDN queries). Given are the three decompounding variants, the stemming only variant, and the baseline (no stemming).

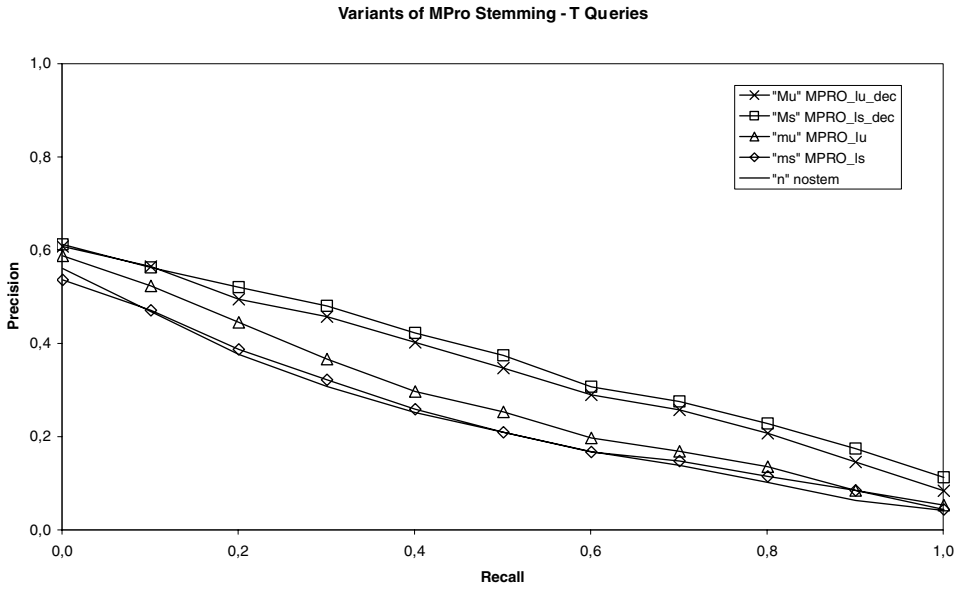

Figure 7. Variants of MPRO stemming (short T queries). Given are the two decompounding variants, the two stemming only variants, and the baseline (no stemming).

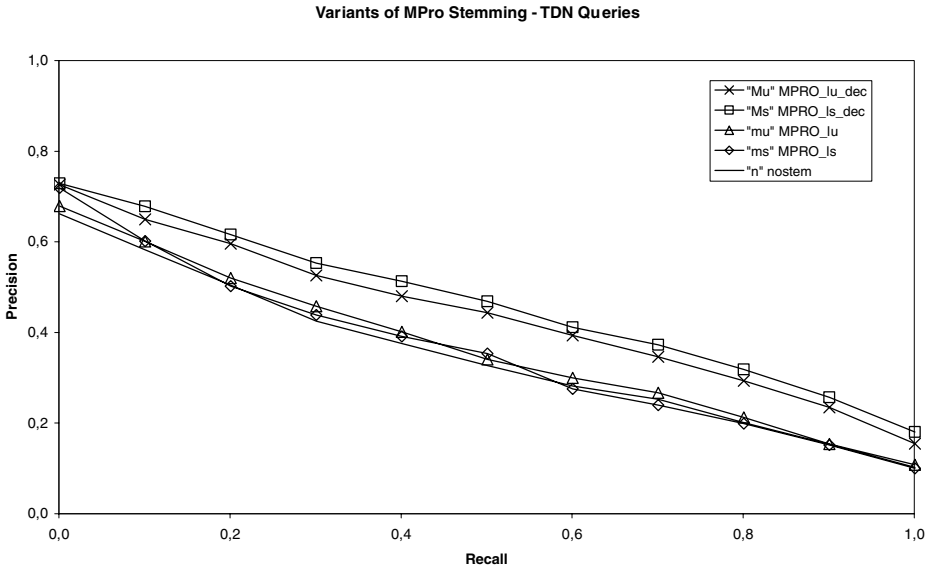

Figure 8. Variants of MPRO stemming (long TDN queries). Given are the two decompounding variants, the two stemming only variants, and the baseline (no stemming).

Looking at the total number of relevant documents retrieved (Table 3), all methods outperform no stemming for both query lengths. Again, the decompounding methods, such as the Spider and MPRO variants and "T" "nist_dec" (upper half of Table 3) perform best, with Spider and MPRO being very close. In this statistic, the combined 6-gram/word-based

Table 3. Number of relevant documents retrieved (total number of relevant documents: 1790).

Run (T)	Recall (T)	Run (TDN)	Recall (TDN)
"Sf" spider_FS	1669 (+30.3%)	"Ms" MRPO_ls_dec	1704 (+11.0%)
"Sc" spider_CS	1665 (+30.0%)	"Mu" MPRO_lu_dec	1696 (+10.5%)
"Ss" spider_SS	1657 (+29.4%)	"Ss" spider_SS	1694 (+10.4%)
"Ms" MPRO_ls_dec	1625 (+26.9%)	"Sc" spider_CS	1693 (+10.3%)
"Mu" MPRO_lu_dec	1624 (+26.8%)	"Sf" spider_FS	1690 (+10.1%)
"T" nist_dec	1577 (+23.1%)	"T" nist_dec	1632 (+6.3%)
"6" 6gram + word	1551 (+21.1%)	"6" 6gram + word	1594 (+3.8%)
"sn" spider_NS	1434 (+11.9%)	"sn" spider_NS	1595 (+3.9%)
"mu" MPRO_lu	1433 (+11.9%)	"t" nist_stem	1595 (+3.9%)
"t" nist_stem	1427 (+11.4%)	"ms" MPRO_ls	1568 (+2.1%)
"ms" MPRO_ls	1398 (+9.1%)	"mu" MPRO_lu	1547 (+0.8%)
"l" linguistica	1300 (+1.5%)	"l" linguistica	1538 (+0.2%)
"n" nostem	1281	"n" nostem	1535

run favorably compares to the stemming methods that do not use decompounding (lower half of Table 3). Of the latter methods, "sn" "spider_NS" performs best.

Recently, "high precision measures", i.e. indications of how well retrieval systems perform if only a small number of highly ranked documents are returned, have become increasingly popular. One main reason for this popularity is the claim that typical web users rarely look past the initial 10 or 20 retrieved items. We use the "Precision @ 10 Documents" measure to investigate if stemming helps in such scenarios. As we indicated in the introduction, stemming is traditionally often seen as a vehicle to increase recall. However, recall plays no important role in a scenario restricted to such few retrieved items. By adopting this precision-centered measure we can observe if stemming approaches help to obtain a better weighting of search terms and therefore a better ranking for the top retrieved items. When using the "Precision @ 10 Documents" measure, differences will become most prominent if queries have more than 10 relevant documents. In case there are very few relevant documents for a query, two systems may obtain the same score even though they retrieve those documents at different ranks. We feel that this is not necessarily a drawback of the measure when only 10 documents are used, as long as these implications are kept in mind. These considerations may become more important, however, when similar scores for larger document sets (e.g. "Precision @ 100 documents") are calculated.

When we look at the results (see Table 4), it becomes clear that a positive effect is most easily seen for the short T queries. There, stemming seems to provide a large number of extra term matches that indeed positively influences ranking. As for average precision, decompounding approaches exhibit the best performance, with the Spider and MPRO variants producing almost identical scores. The effect is less pronounced for long TDN queries, where no stemming outperforms a number of stemming only approaches (plus the NIST decompounding run and the 6-gram/word-based run). The fact that the longer queries often

Table 4. Precision @ 10.

Run (T)	Prec @ 10 (T)	Run (TDN)	Prec @ 10 (TDN)
"Sf" spider_FS	0.3835 (+30.9%)	"Ss" spider_SS	0.4435 (+13.9%)
"Ss" spider_SS	0.3824 (+30.6%)	"Sf" spider_FS	0.4388 (+12.7%)
"Sc" spider_CS	0.3824 (+30.6%)	"Sc" spider_CS	0.4341 (+11.5%)
"Ms" MPRO_ls_dec	0.3800 (+29.7%)	"Ms" MPRO_ls_dec	0.4306 (+10.6%)
"Mu" MPRO_lu_dec	0.3706 (+26.5%)	"Mu" MPRO_lu_dec	0.4247 (+9.1%)
"T" nist_dec	0.3553 (+21.3%)	"T" nist_dec	0.3800 (−2.4%)
"6" 6gram + word	0.3047 (+4.0%)	"6" 6gram + word	0.3412 (−12.4%)
"sn" spider_NS	0.3400 (+16.1%)	"sn" spider_NS	0.3965 (+1.8%)
"mu" MPRO_lu	0.3271 (+11.7%)	"l" linguistica	0.3847 (−1.2%)
"t" nist_stem	0.3259 (+11.3%)	"mu" MPRO_lu	0.3835 (−1.5%)
"l" linguistica	0.2929 (+0.0%)	"t" nist_stem	0.3835 (−1.5%)
"ms" MPRO_ls	0.2906 (−0.7%)	"ms" MPRO_ls	0.3777 (−3.0%)
"n" nostem	0.2929	"n" nostem	0.3894

already contain various different forms of the key search terms means that stemming adds less novel information to be used by the weighting algorithm. Only the more sophisticated Spider and MPRO decompounding can apparently provide good extra matches.

In the following, we will investigate how well the observations we made in this section hold up when we perform a careful statistical analysis.

8. Statistical analysis

We used the IR-STAT-PAK tool by Blustein (1998) for a statistical analysis of the results in terms of average precision, which has been tailored for evaluating multiple "TREC-style" experimental results, supporting a wide range of different popular effectiveness measures. This tool provides an Analysis of Variance (ANOVA), which is the parametric test of choice in such situations but requires checking some assumptions concerning the data. Hull (1993) provides details of these; in particular, the scores in question should be approximately normally distributed and their variance has to be approximately the same for all runs. IR-STAT-PAK uses the Hartley test to verify the equality of variances, and in our case, indicates that the assumption is satisfied, which prompted us to continue with applying the ANOVA test for our data. The program also offers an arcsine transformation, $f(x) = \arcsin(\sqrt{x})$, which Tague-Sutcliffe (1997) suggests for use with Precision/Recall measures to better meet the demand for normally distributed data. We used both raw and transformed scores to assure the suitability of our data for an ANOVA test (see Table 6). Alternatively, non-parametric tests, such as the Friedman test, can be used in case the equality of variance assumption is not satisfied. IR-STAT-PAK does not support non-parametric tests.

Investigating the average precision scores we obtain for T queries, the following results for average precision (after Tukey T test grouping, see Table 5 for ANOVA details) are obtained:

Raw data:

"Sf", "Ss", "Ms", "Sc" $\succ$ "t", "6", "sn", "mu", "ms", "l", "n"
"Mu" $\succ$ "6", "sn", "mu", "ms", "l", "n"
"T" $\succ$ "ms", "l", "n"

Table 5. ANOVA of average precision (T queries).

Experiment		T-raw		T-arcsine		
Source	DF	Mean sq	F	Mean sq	F	Critical
Runs	12	0.232	17.577	0.433	17.393	1.7618
Query	84	0.857	64.880	1.428	57.382	1.2818
Error	1008	0.013		0.025		
Total	1104					

Table 6. ANOVA of average precision (TDN queries).

Experiment		TDN-raw		TDN-arcsine		
Source	DF	Mean sq	F	Mean sq	F	Critical
Runs	12	0.202	12.801	0.365	13.749	1.7618
Query	84	0.895	56.628	1.454	54.852	1.2818
Error	1008	0.016		0.027		
Total	1104					

Arcsine-transformed data:

"Sf", "Ss", "Sc", "Ms" $\succ$ "6", "t", "sn", "mu", "ms", "l", "n"
"Mu" $\succ$ "sn", "mu", "ms", "l", "n"
"T" $\succ$ "ls", "l", "n"

The "$\succ$" symbol denotes a statistically significant difference between two runs with probability $p = 0.95$.

The analysis thus indicates that the all Spider decompounding variants and the respective "Ms" "MPRO_ls_dec" variant significantly outperform all methods without decompounding. They also outperform "Mu" "MPRO_lu_dec" and "T" "nist_dec", but that difference is not statistically significant.

For TDN queries, the findings remain constant. There is a slight shift in ranking of the individual methods, but essentially the same significant differences are found (see also Table 6 for ANOVA details):

TDN queries, raw:

"Sf", "Ms", "Ss", "Sc" $\succ$ "t", "sn", "mu", "l", "ms", "n", "6"
"Mu" $\succ$ "mu", "l", "ms", "n", "6"
"T" $\succ$ "n", "6"

TDN queries, arcsine:

"Ms", "Sf", "Ss", "Sc" $\succ$ "t", "sn", "mu", "ms", "l", "n", "6"
"Mu" $\succ$ "mu", "ms", "l", "n", "6"
"T" $\succ$ "l", "n", "6"

The results for the high-precision scores are very interesting, since we detected less pronounced performance differences than for the average precision scores.

For short T queries, the Spider and MPRO variants using decompounding outperform most stemming only variants (and the baseline of not using stemming at all) to a degree that is statistically significant (see also Table 7 for ANOVA details). This is an important finding,

Table 7. ANOVA of Precision @ 10 (T queries).

Experiment		T-raw		T-arcsine		
Source	DF	Mean sq	F	Mean sq	F	Critical
Runs	12	0.119	10.168	0.281	10.640	1.7618
Query	84	1.308	112.200	2.515	95.228	1.2818
Error	1008	0.012		0.026		
Total	1104					

given that stemming is still often seen mainly as a recall-enhancing device for languages such as English. Stemming can thus be valuable even in cases where recall is of secondary importance, such as often encountered by Web-based search services.

Raw data:

"Sf" ≻ "mu", "t", "6", "n", "l", "ms"
"Sc", "Ss" ≻ "t", "6", "n", "l", "ms"
"Ms", "Mu" ≻ "6", "n", "l", "ms"
"T" ≻ "n", "l", "ms"

Arcsine-transformed data:

"Ss", "Sc" ≻ "t", "mu", "6", "l", "ms", "n"
"Sf", "Ms", "Mu" ≻ "6", "l", "ms", "n"
"T" ≻ "l", "ms", "n"

For long TDN queries, the benefit of stemming and decompounding is less pronounced due to the "richer" representation of the original, unstemmed query when compared to its shorter counterpart (see also Table 8 for ANOVA details). Consequently, fewer significant differences are detected, and no approach or variant outperforms the baseline of not using stemming at all in this scenario.

Table 8. ANOVA of Precision @ 10 (TDN queries).

Experiment		TDN-raw		TDN-arcsine		
Source	DF	Mean sq	F	Mean sq	F	Critical
Runs	12	0.081	5.209	0.161	5.343	1.7618
Query	84	1.215	78.563	2.268	75.044	1.2818
Error	1008	0.015		0.030		
Total	1104					

Raw data:

"Ss" ≻ "ms", "6"
"Sf", "Sc", "Ms", "Mu" ≻ "6"

Arcsine-transformed data:

"Ss", "Ms", "Sf", "Sc", "Mu" ≻ "6"

9. Query-by-query analysis

In addition to the overall analysis presented in Sections 7 and 8, we also analyzed the performance of individual queries in more detail. A set of nine queries (10% of the overall query set) were selected for showing conspicuous behavior, either in terms of overall performance or because of outliers by individual methods. Various measures, such as mean average precision, high precision measures (mean of precision at 5, 10, 15 documents retrieved, mean of precision at 10, 20, 30 documents retrieved, and precision at 100 documents retrieved) and uniquely retrieved relevant documents, as well as their respective standard deviations, were used to identify this set of interesting queries. For each query in this set, we determined the most important keywords, and examined all methods with regard to the stems that they produce, and the constituents of the compounds as identified by the various decompounding approaches.

We continue by giving a short analysis of each of the nine selected queries. The title fields of the nine queries are given as heading to each subsection. Key search terms mentioned in the analysis can be contained in any of the fields of the query (title, description, narrative).

9.1. CLEF 2000 Campaign, Topic 1: "Architektur in Berlin (Architecture in Berlin)"

Approaches performing well: "Sf", "Sc", "Ss", "Ms", "l"

The key search terms in this query are "Architektur" (architecture) and "Berlin". The best results regarding average precision, early precision, and number of relevant documents retrieved come from the Spider decompounding runs ("Sf", "Sc", "Ss"), the "Ms" "MPRO_ls_dec" decompounding run, and Linguistica. Common characteristic of these runs is that they conflate "Architektur" (architecture) and "Architekt" (architect), which pulls in approximately 10% more relevant documents than the methods that omit this conflation.

9.2. CLEF 2000 Campaign, Topic 5: "Mitgliedschaft in der Europäischen Union (European Union Membership)"

Approaches performing well: "n", "l", "t", "sn", "mu", "Mu"

Key search terms in this query are "Mitgliedschaft/Mitgliedstaaten/Mitgliedsstaaten" (membership/member state) and "Europäischen" (European).

A peculiarity of this query is the two different spellings for "Mitglied(s)staaten", once with binding-s (in the "description" part), and once without (in the "narrative" part). While probably not intended by the topic creator, this is not strictly speaking a typo, since both forms are valid. The "Sf", "Sc", "Ss", "ms", "Ms" and "Mu" methods conflate the two variations, whereas the other methods produce two distinct indexing features. However, conflation of the two different spellings for "Mitglied(s)staaten" does not seem to be essential for good retrieval.

Generally speaking, decompounding seems to be counterproductive for this query. Apart from the exception of "Mu", all decompounding methods produce results that are substantially worse (in the case of the Spider stemmer, between 48% and 66%) than the ones returned by methods without decompounding (or even without stemming at all). It seems that the core concept of "member state" gets diluted too much when its constituents are added as independent search terms. From a linguistic point of view, "Mitgliedschaft" should not be split, contrary to the behavior of some of the approaches, since "schaft" is in this case a derivational suffix.

9.3. CLEF 2000 Campaign, Topic 26: "Nutzung von Windenergie (Use of Wind Power)"

Approaches performing well: all decompounding approaches.

Key search term is "Windenergie" (wind power).

Methods using decompounding outperform other methods substantially. Decomposition of "Windenergie" into "Wind" and "Energie" results in the retrieval of nearly twice as many relevant documents. Stemming proper makes little difference, since the keywords occur in their base forms in the query statement and in the relevant documents.

9.4. CLEF 2000 Campaign, Topic 34: "Alkoholkonsum in Europa (Alcohol Consumption in Europe)"

Approaches performing well: all decompounding approaches.

Key search terms are "Alkoholkonsum/Alkoholmissbrauch" (alcohol consumption/alcohol abuse) and "Europa" (Europe). This is another query where decompounding is crucial. Methods with decompounding (most important to find other compounds consisting of "alkohol" as constituent) retrieve 50% additional relevant documents on average.

9.5. CLEF 2000 Campaign, Topic 35: "Wölfe in Italien (Wolves in Italy)"

Approaches performing well: no clear picture

Keywords are "Wölfe" (wolves) and "Italien/Italiens" (Italy, of Italy). This query is a special case: it has only one relevant document in the collection. Stemming differences across methods are minor, mainly in terms of some less frequent word forms being conflated or not. All methods find the one relevant document, but average precision numbers fluctuate wildly depending on the rank at which the document is retrieved. These large fluctuations are mainly due to the properties of the average precision measure, and we believe them to obscure the performance of the different stemming methods in this case.

9.6. CLEF 2001 Campaign, Topic 6: "Embargo gegen den Irak (Embargo on Iraq)"

Approaches performing well: "n", "l"

Key search terms are "Embargo" (embargo), "Irak/irakischen" (Iraq, Iraqi) and "Bevölkerung" (population). The best two methods for this query are "no stemming" and "Linguistica", which both do no or very little stemming. This is apparently due to the fact that all keywords are present in their base forms in the query statement as well as in the documents, with stemming only introducing further noise.

9.7. CLEF 2001 Campaign, Topic 21: "Ölkatastrophe in Sibirien (Siberian Oil Catastrophe)"

Approaches performing well: "Sf", "Sc", "Ss"

Keywords are "Ölkatastrophe/Ölpipeline/Ölleitung" (oil catastrophe/oil pipeline/oil pipeline), "Umweltaspekte" (environmental aspects), "Ökosystem" (ecological system) and "Sibiren/Sibiriens" (Siberia/of Siberia). The Spider decompounding methods are the most aggressive in producing conflations: they are the only methods that decompound all the compound keywords except "Ökosystem", which is not decompounded by any method. MPRO splits "Oelleitung" but not "Ölleitung" because of the umlaut treatment in the version we used.

In general, this query demonstrates how decompounding helps to absorb differences in writing styles. For instance, the query uses "Ölpipeline" in the description part and "Ölleitung" in the narrative part of the query, which are essentially synonyms.

9.8. CLEF 2002 Campaign, Topic 25: "Schatzsucher (Treasure Hunting)"

Approaches performing well: decompounding methods and "t"

Key search terms are "Schatzsucher/Schatzsuche" (treasure hunter/treasure hunting), "Reliquien" (relics), "Schiffen" (ships).

To achieve high performance, it seems to be essential to establish a relation between "Schatzsucher" and "Schatzsuche". The decompounding methods do this via the component "Schatz" (treasure), whereas "t" "NIST_stem" conflates the two words to a common stem ("Schatzsu"). All other methods that fail to establish such a relation perform substantially worse. Another nicety of this query is the accidental conflation of "finde" (look for) and "Fund" (a find) to "find" by the Spider variations which positively affects the result.

9.9. CLEF 2001 Campaign, Topic 27: "Schiffskollisionen (Ship Collisions)"

Approaches performing well: "Sf", "Sc", "Ss"

Key search terms are "Schiffskollisionen" (ship collisions), together with its constituents "Kollisionen" (collisions), "Schiffen" (ships) and "Zusammenstößen" (collisions) in the narrative part of the query. The best results come from the Spider variations that use decompounding. These are the only methods that split "Schiffskollisionen", which seems to be essential for good retrieval performance for the short T version of this query. In the TDN

run the constituents occur as separate words resulting in much better performance for all systems except for no stemming and Linguistica.

In summary, we found that stemming is in most cases beneficial independent of the method applied. Approaches conducting a balanced conflation, i.e. avoiding both understemming and overstemming as best as possible, seem to be superior.

Cases where no stemming outperforms most or all of the stemming methods exist when all the keywords are already present in their base forms (and occur rarely in inflected forms) and represent simple concepts (no compound nouns) (e.g. Topic 6, 2001 Campaign). Furthermore, queries consisting of proper names show such effects, especially in short queries. For long queries, in these cases, the noise introduced by stemming cancels out the benefits from matching additional word forms.

Not surprisingly, we found a number of queries where the key to successful retrieval is clearly the ability to split compounds, since the constituents are important to the respective topic statements, and the compound commonly is transcribed in phrasal form in the documents.

However, there are also cases where decompounding is counterproductive (such as topic 5, 2000 campaign). Thus, it is important to avoid the decomposition of "false" compounds such as "Mitgliedschaft" (membership) and of compounds for which the splitting causes a shift of meaning such as "Frühstück" (breakfast), composed of "Früh" (early) and "Stück" (piece). Also, decompounding of words that are rarely written in phrasal form e.g. "Mitgliedstaaten" (member states) may negatively impact weighting.

There may be potential in development of corpus-based decompounding methods in this regard, as it may well depend on the document collection itself if a key concept of the query gets diluted by compound splitting, or if additional, good matches are found. An interesting line of research may be to consider how to automatically infer from corpus statistics how well compounds are represented by their constituents, and use this as a factor in the decision of whether to apply compound splitting or not. Similar considerations apply for stemming proper, where especially the effect of handling derivational suffixes may differ depending on the terms or even the query and collection used for searching.

Lastly, we found some oddities, which were either due to the evaluation measure used (e.g. mean average precision in case that there are very few relevant documents for a query) or to some erroneous conflations, which turned out to produce good matches by chance.

10. Conclusions

In our experiments, we demonstrated that stemming is useful for German text retrieval in most cases. Compared to a system without stemming, we obtained performance gains measured in mean average precision of up to 23% for short (T) and up to 11% for long queries (TDN). For recall we observed improvements of 12% for T, and 4% for TDN. Exceptions where stemming is detrimental are queries comprising proper names or other invariant words (where no inflection exists, or the inflection is rarely used). The different stemming methods show only weak significant differences, mainly in the high recall range.

An important finding of this study is that decompounding contributes more to performance improvement than stemming, 16% to 34% for short and 9% to 28% for long queries. In

contrast to stemming, we observed larger differences between individual methods implying that it is important to carefully choose the right degree of decompounding. There are two main goals with respect to decompounding: Firstly to produce as many extra matches for retrieval as possible, and secondly to avoid inappropriate splittings. The two best runs, "Sf" "spider_FS" and "Ms" "MPRO_ls_dec", put different emphasis on these goals. "Sf" "spider_FS" splits aggressively, and "Ms" "MPRO_ls_dec" avoids linguistically incorrect splittings. However, they show almost the same performance.

Looking beyond the popular average precision measure, we have investigated the impact of stemming in a "high precision scenario", such as often assumed in Web settings. Traditionally, stemming is often seen mainly as a vehicle to enhance recall. However, our experiments indicate the best stemming and decompounding approaches lead to better precision at 10 documents retrieved than using no stemming. Some of the differences are statistically significant.

This report makes a major contribution to the discussion on the benefits of stemming that goes beyond the focus on the German language by employing a complete spectrum of stemming and decompounding approaches on a large reliable corpus, which allowed us to conduct a thorough analysis of the results. One outcome of such a broad analysis is that stemming can be done using comparatively simple approaches such as the NIST German stemmer, which showed competitive performance. In contrast, decompounding requires a more sophisticated approach, either providing a sufficient lexical coverage for aggressive splitting or linguistic knowledge to achieve correct handling of compounds.

Furthermore, we found that for German as a morphological rich language, purely language independent methods do not give competitive performance. The exception is the 6-gram/word-based run, but only for short queries. Even so, the results are still significantly worse than for other methods using decompounding, and the increased index size from using n-grams is a substantial drawback.

Future work could validate these findings that we obtained for unrestricted text for settings where domain specific terminology is frequent. Such terminology can have properties that may necessitate adaptations especially related to decompounding (e.g. medical terms of Latin origin, or names of chemical substances).

Acknowledgments

We would like to thank IAI for the opportunity to use MPRO. Thanks go to John Goldsmith for providing the Linguistica software, and the CLEF consortium and its data providers for the construction of the test collection. Decompounding for the NIST stemmer is based joint work with Paul Over from NIST. Jacques Savoy and three anonymous referees provided detailed comments and suggestions that helped improve the paper.

Note

1. Topics are "statements of user needs". They form the basis for the formulation of queries, which are then run against the document collection. In all our experiments, we constructed the queries without manual intervention ("automatic experiments"), by indexing all or part of the topic text.

References

Blustein J (1998) IR STAT PAK. URL: http://www.csd.uwo.ca/∼jamie/IRSP-overview.html (last visit 11/19/2002).

Braschler M and Schäuble P (2001) Experiments with the Eurospider retrieval system for CLEF 2000. In: Peters C, Ed., Cross-Language Information Retrieval and Evaluation, Workshop of the Cross-Language Evaluation Forum, CLEF 2000, pp. 140–148.

Choueka Y (1992) Responsa: An operational full-text retrieval system with linguistic components for large corpora. In: Computational Lexicology and Lexicography: A Volume in Honor of B. Quemada.

Frakes WB (1992) Stemming Algorithms. In: Frakes WB and Baeza-Yates R, Eds., Information Retrieval, Data Structures & Algorithms. Prentice Hall, Eaglewood Cliffs, NJ, USA, pp. 131–160.

Frisch E and Kluck M (1997) Pretest zum Projekt German Indexing and Retrieval Testdatabase (GIRT) unter Anwendung der Retrievalsysteme Messenger und freeWAISsf. IZ Arbeitsbericht Nr. 10, GESIS IZ Soz., Bonn, Germany (in German).

Goldsmith J (2001) Unsupervised learning of the morphology of a natural language. Computational Linguistics, 27(2):153–198. MIT Press. Available at: URL: http://humanities.uchicago.edu/faculty/goldsmith/Linguistica2000/ (last visit 11/19/2002).

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

Harman D (1997) The TREC conferences. In: Sparck-Jones K and Willett P, Eds., Readings in Information Retrieval. Morgan Kaufmann Publishers, San Francisco, CA, USA.

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

Hull DA (1993) Using statistical testing in the evaluation of retrieval experiments. In: Proceedings of the 16th ACM SIGIR Conference. Pittsburg, USA.

Hull DA, Grefenstette G, Schultze BM, Gaussier E, Schütze H and Pedersen O (1996) Xerox TREC-5 site report: Routing, filtering, NLP and Spanish tracks. In: Proceedings of the Fifth Text Retrieval Conference (TREC 5). Gaithersburg, USA.

Kraaij W and Pohlmann R (1996) Using linguistic knowledge in information retrieval. OTS Working Paper OTS-WP-CL-96-001, University of Utrecht, The Netherlands.

Kraaij W and Pohlmann R (1996) Viewing stemming as recall enhancement. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Zurich, Switzerland.

Krause J and Womser-Hacker C (1990) Das Deutsche Patent-informationssystem. Entwicklungstendenzen, Retrievaltests und Bewertungen. Carl Heymanns (in German).

Lovins JB (1968) Development of a stemming algorithm. Mechanical Translation and Computational Linguistics, 11(1/2):22–31.

Maas D (1996) MPRO—Ein System zur Analyse und Synthese deutscher Wörter. In: Hauser R, Ed., Linguistische Verifikation, Max Niemeyer Verlag, Tübingen (in German).

Mayfield J, McNamee P and Piatko C (1999) The JHU/APL HAIRCUT system at TREC-8. In: Proceedings of the Eighth Text REtrieval Conference (TREC-8). NIST Special Publication 500-246, pp. 445–451.

Monz C and de Rijke M (2002) Shallow morphological analysis in monolingual information retrieval for Dutch, German and Italian. In: Peters C, Braschler M, Gonzalo J, and Kluck M, Eds., Evaluation of Cross-Language Information Retrieval Systems. CLEF 2001, Lecture Notes in Computer Science, LNCS 2406, pp. 262–277.

Moulinier I, McCulloh JA and Lund E: West Group at CLEF 2000 (2001) Non-English monolingual retrieval. In: Peters C, Ed., Cross-Language Information Retrieval and Evaluation, Workshop of the Cross-Language Evaluation Forum, CLEF 2000, pp. 253–260.

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, 3(5):384–390.

Porter MF (1980) An algorithm for suffix stripping. Program, 14(3):130–137. Reprint in: Sparck Jones K and Willett P., Eds., Readings in Information Retrieval, Morgan Kaufmann Publishers, San Francisco, CA, USA, pp. 313–316.

Ripplinger B (2002) Linguistic knowledge in cross-language information retrieval. PhD Thesis, Herbert Utz Verlag, Munich, Germany.

Savoy J (1999) A stemming procedure and stopword list for general French corpora. Journal of the American Society for Information Science, 50(10):944–952.

Savoy J (2003) Cross-language information retrieval: Experiments based on CLEF 2000 Corpora. Information Processing & Management, 39(1):75–115.

Singhal A, Buckley C and Mitra M (1996) Pivoted document length normalization. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Zurich, Switzerland.

Sheridan P and Ballerini JP (1996) Experiments in multilingual information retrieval using the SPIDER system. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Zurich, Switzerland.

Tague-Sutcliffe J (1997) The pragmatics of information retrieval experimentation, revisited. In: Sparck-Jones K and Willett P, Eds., Readings in Information Retrieval. Morgan Kaufmann Publishers, San Francisco, CA, USA.

Tomlinson S (2002) Stemming evaluated in 6 Languages by Hummingbird SearchServer™ at CLEF 2001. In: Peters C, Braschler M, Gonzalo J and Kluck M, Eds., Evaluation of Cross-Language Information Retrieval Systems. CLEF 2001, Lecture Notes in Computer Science, LNCS 2406, pp. 278–287.

Wechsler M, Sheridan P and Schäuble P (1997) Multi-language text indexing for internet retrieval. In: Proceedings of the 5th RIAO Conference, Computer-Assisted Information Searching on the Internet, Montreal, Canada, pp. 217–232.

Womser-Hacker C (1989) Der PADOK-Retrievaltest. In: "Sprache und Computer" Band 10, Georg Olms Verlag (in German).