Published in Journal of the American Society for Information Science and Technology, 60, issue 12, 1616-1624, 2009
which should be used for any reference to this work

Algorithmic Stemmers or Morphological Analysis? An Evaluation

Claire Fautsch and Jacques Savoy
Computer Science Department, University of Neuchâtel, 2009 Neuchâtel, Switzerland.
E-mail: {Claire.Fautsch, Jacques.Savoy}@unine.ch

It is important in information retrieval (IR), information extraction, or classification tasks that morphologically related forms are conflated under the same stem (using stemmer) or lemma (using morphological analyzer). To achieve this for the English language, algorithmic stemming or various morphological analysis approaches have been suggested. Based on Cross-Language Evaluation Forum test collections containing 284 queries and various IR models, this article evaluates these word-normalization proposals. Stemming improves the mean average precision significantly by around 7% while performance differences are not significant when comparing various algorithmic stemmers or algorithmic stemmers and morphological analysis. Accounting for thesaurus class numbers during indexing does not modify overall retrieval performances. Finally, we demonstrate that including a stop word list, even one containing only around 10 terms, might significantly improve retrieval performance, depending on the IR model.

Introduction

Stemming refers to the conflation of word variants into a common stem (or form when the string cannot be found in the language). In information retrieval (IR), the application of a stemming procedure when indexing documents or requests is assumed to be a good practice (Manning, Raghavan, & Schütze, 2008), although the *N*-gram indexing strategy is typically an exception to this rule (McNamee & Mayfield, 2004). For example, when a query contains the word "horse," it seems reasonable also to retrieve documents containing the related word "horses," a practice which usually tends to improve retrieval effectiveness. Designing effective stemming procedures also may be helpful for other purposes, such as text mining, natural language processing, or gathering statistics on a document corpus.

For the English language, various authors have proposed algorithmic stemmers based on the morphological rules of this language (e.g., Lovins, 1968; Porter, 1980). An alternative is to apply a more complex morphological analysis requiring additional computational resources and a dictionary able to return the correct lemma (or dictionary entry). Moreover, as a means of defining better matches between terms occurring in the query and the document, we also might make use of part-of-speech (POS) information (Krovetz, 1993; Savoy, 1993). Finally, once a word's corresponding lemma has been found, we also could consider the word's various synonyms, making use of synset numbers (i.e., thesaurus class number) available in the WordNet thesaurus (Fellbaum, 1998).

The main objective of this article is to analyze and evaluate various stemming strategies using a relatively large number of queries. On the other hand, a morphological analysis can be applied to return the lemma for each surface word. This second word-normalization strategy also will be evaluated in this study. The rest of the article is organized as follows: First, we describe related stemming approaches and then depict the main characteristics of our test collection. In the subsequent section, we briefly describe the IR methods applied during our experiments. We then evaluate the performance of various IR models along with different algorithmic stemmers or morphological analysis, and also analyze the use of POS information and thesaurus class numbers. Finally, we present the main findings of this article.

Related Work

In the IR domain, stemming is usually considered as an effective means of enhancing retrieval performance through conflating several different word variants into a common form. As a first approach to designing a stemmer, we begin by removing only inflectional suffixes so that singular and plural word forms (e.g., "dogs" and "dog") or feminine and masculine variants (e.g., "actress" and "actor") will conflate to the same root. Suffix removal also is controlled through the adjunct of quantitative restrictions (e.g., "ing" would be removed if the resulting stem had more than three letters, as in "running," but not in "king") or qualitative restrictions

(e.g., "-ize" is removed if the resulting stem does not end with "e" as in "seize"). Moreover, certain ad hoc rules are used to correct spelling and improve conflation accuracy (e.g., "running" becomes "run," and not "runn"), due to certain irregular grammar rules usually applied to a language to facilitate pronunciation. Of course, one should not stem proper nouns such as "Collins" or "Hawking," at least when the system can recognize them.

These suffix-removal methods are based on a set of rules known as algorithmic stemmers, and thus ignore word meaning and POS categories. Other stemming techniques that remove only morphological inflections are termed "light" suffix-stripping algorithms, such as the S-stemmer (Harman, 1991), and apply three rules to remove the plural morpheme "-s." There also are more sophisticated approaches that remove derivational suffixes (e.g., "-ment," "-ably," "-ship" in the English language). Those suggested by Lovins (1968) are based on a list of over 260 suffixes whereas Porter's (1980) algorithm looks for about 60 suffixes.

Stemming methods are usually designed to work with general texts and work with any given language; however, certain stemming procedures may be designed especially for a specific domain (e.g., medicine) or a given document collection, such as that of Xu and Croft (1998). They suggested developing stemming procedures using a corpus-based approach which more closely reflects the language used (including word frequencies and other co-occurrence statistics) instead of using a set of morphological rules in which the frequency of each rule (and therefore its underlying importance) is not precisely known.

Algorithmic stemming procedures tend to make errors, usually due to overstemming (e.g., "general" becomes "gener," and "organization" is reduced to "organ") or to understemming [e.g., with Porter's (1980) stemmer, the words "create" and "creation" or "European" and "Europe" do not conflate to the same root]. In general, however, stemming tends to improve recall, yet these examples show it also may decrease precision, rendering Web search strategies problematic. In this case, Peng, Ahmed, Li, and Lu (2007) suggested applying context-sensitive stemming methods to search terms based on a statistical language model. Krovetz (1993) suggested another method of reducing stemming errors involving an online dictionary to produce better conflations.

As an alternative approach that requires a deeper morphological knowledge, however, we can apply a more complex morphological analysis capable of precisely defining the corresponding lemma (or entry in the dictionary) for each inflected word form. Flexions can thus be removed to obtain the lemma (e.g., "houses" becomes "house"), and the resulting POS information could be used to further enhance the quality of the suffix-removal process. For example, the derivational suffix "-able" is used to form an adjective from a verb stem as in "readable" or "thinkable." Such a process is applied for the French language (Savoy, 1993). Such morphological analysis is used more frequently for languages that have a complex morphology, such as Finnish, Swedish, German, or Russian (Kettunen, 2007; Tomlinson, 2004).

Based on an analysis of IR stemming performances, Harman (1991) demonstrated that no statistically significant improvement would result from applying three different algorithmic stemmers; namely, that of Lovins (1968), Porter (1980), and the light S-stemmer (that conflates only singular and plural English word forms). A query-by-query analysis revealed that stemming did affect the performance, with the number of queries showing improved performance almost equaling the number of queries showing poorer performance. Other studies (e.g., Hull, 1996) based on a single IR model (a variant of the classical *tf–idf* method) have shown that stemming resulted in modest improvements, ranging from 1 to 3%. This analysis revealed, however, that stemming tends to make a difference for many individual queries. According to Hull's (1996) study, all stemmers resulted in statistically superior average precision than did a nonstemming approach. Moreover, the S-stemmer proved to be less effective than either the Lovins or Porter methods.

Based on these facts, the rest of this article will address the following questions: (a) With a large set of queries (∼300), is suffixing really better than a nonstemming approach? (b) Is it possible to obtain improved retrieval effectiveness when applying a morphological analysis instead of an algorithmic stemmer? (c) Is it possible to obtain statistically significant differences between various algorithmic stemmers? (d) Does the use of thesaurus class numbers or simple POS information prove useful in increasing retrieval effectiveness?

Test Collections

The evaluations reported in this article were based on the English test collections built during the Cross-Language Evaluation Forums (CLEF) 2001 through 2006 evaluation campaigns (Peters et al., 2008) and regrouped into the Robust track in CLEF 2008. This corpus consists of articles published in 1994 in the *Los Angeles Times* as well as others extracted from the *Glasgow Herald* newspapers published in 1995. This collection contains a total of 169,477 documents (∼579 MB of data), and each article contains about 250 on average ($Mdn = 191$) content-bearing terms (not counting commonly occurring words such as "the," "of," or "in"). Typically, documents in this collection are represented by a short title plus one to four paragraphs of text, and both American and British English spellings can be found in the corpus.

This collection contains 310 topics, each subdivided into a brief title (denoted as *T*), a full statement of the information need (called the *description* or *D*), and any background information that might help assess the topic (the *narrative* or *N*) (see Table 2). These topics cover various subjects (e.g., "El Niño and the Weather," "Chinese Currency Devaluation," "Eurofighter," "Victories of Alberto Tomba," "Marriage Jackson–Presley," or "Computer Animation"), including both regional ("Films Set in Scotland," "Area of Kaliningrad") and international coverage ("Oil Prices," "Sex in Advertisements"). In our evaluations, we built the queries based on the

TABLE 1. A few CLEF test-collections statistics.

	2001	2002	2003	2004	2005	2006
Source	*LA Times*	*LA Times*	*LA Times* *Glasgow Herald*	*Glasgow Herald*	*LA Times* *Glasgow Herald*	*LA Times* *Glasgow Herald*
Size (MB)	425	425	579	154	579	579
No. of documents	113,005	113,005	169,477	56,472	169,477	169,477
No. of topics	47	42	54	42	50	49
Topics	41–90	91–140	141–200	201–250	251–300	301–350

TABLE 2. Example of a query with and without lemma, WordNet thesaurus number (synset), and part-of-speech (POS) tag.

<NUM> C062 </NUM>
<EN-TITLE> Northern Japan Earthquake </EN-TITLE>
<EN-DESC> Find documents that report on an earthquake on the east coast of Hokkaido, northern Japan, in 1994. </EN-DESC>
<EN-NARR> Documents describing an earthquake with a magnitude of 7.9 that shook Hokkaido and other northern Japanese regions in October 1994 are relevant. Also of interest are tidal wave warnings issued for Pacific coastal areas of Hokkaido at the time of the earthquake. Documents reporting any other earthquakes in Japan are not relevant. </EN-NARR>
...
<NUM> C062 </NUM>
<EN-TITLE>
<TERM ID="C062-1" LEMA="northern" POS= "NNP;;>
<WF> Northern </WF>
<SYNSET SCORE="1" CODE="05210354-n"/> </TERM >
<TERM ID="C062-2" LEMA="japan" POS= "NNP">
<WF> Japan </WF>
<SYNSET SCORE="0.4451194309593595" CODE="06520317-n"/>
<SYNSET SCORE="0.5548805690406404" CODE="06519251-n"/> </TERM>
<TERM ID="C062-3" LEMA="earthquake" POS="NN">
<WF> Earthquake </WF>
<SYNSET SCORE="1" CODE="05526375-n"/> </TERM>
</EN-TITLE> ...

T and *D* parts of the topic formulation, corresponding to the official query format in the CLEF evaluation campaigns.

Relevance judgments (correct answers) were supplied by human assessors throughout the various CLEF evaluation campaigns. As shown in Table 1, the entire corpus was not used during all the evaluation campaigns, and thus pertinent articles had to be searched in different parts of the corpus. For example, Topics 201 to 250 were created in 2004 and were responses resulting from searches in the *Glasgow Herald* (1995) collection, a subset representing 56,472 documents. Of the 50 queries originally available in 2004, we found that only 42 returned at least one correct answer.

In all, 26 queries were removed because there were no relevant documents in the corpus, meaning only 284 $(310 - 26)$ topics were used in our evaluation. Upon an inspection of these relevance assessments, the average number of correct responses for each topic was 22.46 ($SD = 28.9$, $Mdn = 11.5$), with Topic 254 ("Earthquake Damage") obtaining the greatest number of relevant documents (229).

During the Robust track at CLEF 2008, the organizers also provided an extended version of both documents and topic descriptions (see the bottom part of Table 2) with additional information that could be used to verify whether word-sense disambiguation (WSD) might improve retrieval effectiveness. To achieve this objective, each surface word (after the label <WF>) was preceded by its corresponding lemma (under the tag <TERM>, a value placed after the keyword LEMMA) with its corresponding POS tag. The latter information was given according to a variant of the Penn Treebank tag set (Marcus, Santorini, & Marcinkiewicz, 1993). As seen in our example, the tag "NN" was used to indicate a noun, and the tag "NNP" indicated a proper noun. With the lemma information, the morphological analysis results become available; therefore, a stemming procedure is no longer needed.

As shown in Table 2, synset information is placed after the string corresponding to the surface word, linking it to the entry in the WordNet thesaurus (Version 1.6) (information given after the tag <SYNSET>). This entry could be unique (as in our example with the word "whale"). For a proper noun (e.g., personal, geographic, or product name), no pertinent entry in the WordNet thesaurus can be found; thus, the corresponding synset information is not given. Finally, a term may belong to different synsets (The noun "reserve" belongs to three synsets.) In such cases, each possible entry is preceded by a probability estimation that the corresponding synset is correct.

Not all of this information is introduced manually. The MXPOST (Maximum Entropy POS Tagger; freely available at http://www.inf.ed.ac.uk/resources/nlp/local_doc/MXPOST.html; Ratnaparkhi, 1996) identifies the POS of

each word, and then the corresponding lemma is extracted using the Java WordNet Library (JWNL), an application programming interface used to provide easy access to the WordNet relational thesaurus. Based on this information along with local collocations and surrounding words, the NUS-PT WSD system (Chan, Ng, & Zhong, 2007) disambiguates the word-type based on the Vector Support Machine (VSM) approach trained with the SemCor corpus, as well as other training examples extracted from parallel texts serving as training data. In a related study with WordNet, Voorhees (1993) attached only a single synset number to each noun (Use the most frequently occurring synset number found in the surrounding text if there are multiple possibilities).

IR Models

To obtain a broader view of the relative performance of different retrieval models with respect to different stemmers and to the morphological analysis, we have implemented five search models. When using this evaluation approach, we want to ground our findings on a more solid basis since a hidden or an unknown property of the collection or a stemmer may favor one model over the other. Grounding on several approaches we will partly resolve this problem.

Following this principle, we first used the classical *tf–idf* model wherein the weight attached to each indexing term was the product of its term-occurrence frequency (tf_{ij} for indexing term t_j in document d_i) and the logarithm of its inverse document frequency [$idf_j = \log(n/df_j)$]. To measure similarities between documents and requests, we computed the inner product after normalizing (cosine) the indexing weights (Manning et al., 2008). This IR model was used in studies by Voorhees (1993) and by Hull (1996), for example.

To complement this vector-space model, we implemented certain probabilistic models such as the Okapi (or BM25) approach (Robertson, Walker, & Beaulieu, 2000), and two models derived from *Divergence from Randomness* (DFR) paradigm (Amati & van Rijsbergen, 2002) wherein the two information measures formulated next are combined:

$$w_{ij} = \mathrm{Inf}^1_{ij} \cdot \mathrm{Inf}^2_{ij} = -\log_2[\mathrm{Prob}^1_{ij}] \cdot (1 - \mathrm{Prob}^2_{ij}) \quad (1)$$

in which Prob^1_{ij} is the probability of finding by pure chance the tf_{ij} occurrences of the term t_j in a document. On the other hand, Prob^2_{ij} is the probability of encountering a new occurrence of term t_j in the document, given that tf_{ij} occurrences of this term already had been found. To calculate these two probabilities, we used the I(n_e)C2 model, based on the following estimates:

$$\mathrm{Prob}^1_{ij} = \left(\frac{n}{n_e + 1}\right)^{tfn_{ij}}$$

$$\text{and } \mathrm{Prob}^2_{ij} = 1 - \left(\frac{tc_j + 1}{df_j \cdot (tfn_{ij} + 1)}\right) \quad (2)$$

$$\text{with} \qquad tfn_{ij} = tf_{ij} \cdot \ln\left(1 + \frac{c \cdot \text{mean dl}}{l_i}\right)$$

$$\text{and} \qquad n_e = n \cdot \left(1 - \left(\frac{n-1}{n_i}\right)^{tc_j}\right)$$

where tc_j is the number of occurrences of term t_j in the collection, df_j indicates the number of documents in which the term t_j occurs, n the number of documents in the corpus, l_i the length of document d_i, *mean dl* (= 212), the average document length, and c a constant (fixed empirically at 1.5).

For our second DFR model, called DFR-PL2, the implementation of Prob^1_{ij} is given by Equation 3, and Prob^2_{ij} is given by Equation 4, as follows:

$$\mathrm{Prob}^1_{ij} = (e^{-\lambda_j} \cdot \lambda_j^{tf_{ij}})/tf_{ij}! \quad \text{with } \lambda_j = tc_j/n \quad (3)$$

$$\mathrm{Prob}^2_{ij} = tfn_{ij}/(tfn_{ij} + 1) \quad (4)$$

Finally, we also applied a language model (LM) approach (Hiemstra, 2000), known as a nonparametric probabilistic model. Within this LM paradigm, various implementation and smoothing methods also might be considered. In this article, we adopted a model proposed by Hiemstra (2000, 2002) as described in Equation 5 and using the Jelinek–Mercer smoothing method (Zhai & Lafferty, 2004), a combined estimate based on both the document ($P[t_j \mid d_i]$) and the entire corpus ($P[t_j \mid C]$).

$$\mathrm{Prob}[d_i|q] = \mathrm{Prob}[d_i] \cdot \Pi_{tj \in Q}[\lambda_j \cdot \mathrm{Prob}[t_j|d_i] + (1 - \lambda_j) \cdot \mathrm{Prob}[t_j|C]]$$

$$\text{with} \qquad \mathrm{Prob}[t_j|d_i] = \left(\frac{tf_{ij}}{l_i}\right)$$

$$\text{and} \qquad \mathrm{Prob}[t_j|C] = \left(\frac{df_j}{lc}\right) \quad \text{with} \quad lc = \sum_{k=1}^{t} df_k \quad (5)$$

where λ_j is a smoothing factor (fixed at 0.35 for all indexing terms t_j), df_j indicates the number of documents indexed with the term t_j, and lc is a constant related to the size of the underlying corpus C.

Evaluation

To measure retrieval performance (Buckley & Voorhees, 2005), we adopted the mean average precision (MAP) computed by TREC_EVAL based on a maximum of 1,000 retrieved items. To statistically determine whether a given search strategy is statistically better than another, we applied the bootstrap methodology (Savoy, 1997), with the null hypothesis H_0 stating that both retrieval schemes produce similar performance. In the experiments presented in this article, statistically significant differences were detected by applying a two-sided test (significance level $\alpha = 5\%$). This null hypothesis would be accepted if two retrieval schemes returned statistically similar means; otherwise, it would be rejected.

TABLE 3. Mean average precision (MAP) of various IR models and different stemmers (284 title/description queries).

	MAP					
	None	S-stemmer	Porter	Lovins	SMART	Lemma
Okapi	**0.4345**	0.4648†	0.4706†	0.4560‡	0.4755†	0.4663†
DFR-PL2	*0.4251*	*0.4553*†	*0.4604*†	0.4499†‡	*0.4634*†	0.4608†
DFR-I(n_e)C2	0.4329	**0.4658**†	**0.4721**†	**0.4565**‡	**0.4783**†	**0.4671**†
LM	*0.4240*	*0.4493*†	*0.4555*†	*0.4389*‡	*0.4568*†	*0.4444*†
tf– idf	*0.2669*	*0.2811*†	*0.2839*†	*0.2650*‡	*0.2860*†	*0.2778*†
Average	0.4291	0.4588	0.4647	0.4503	0.4685	0.4597
%change		+6.9	+8.3	+4.9	+9.2	+7.1

IR Models Evaluation

Based on the methodology previously described, the MAP obtained from applying six stemming approaches to five IR models is shown in Table 3. The second column (labeled *None*) lists the retrieval performances obtained when ignoring the stemming stage during the indexing or query processing. The "S-stemmer" column lists the performance obtained by the light stemmer based on three rules (Harman, 1991), and the MAP obtained by either Porter's (1980) or Lovins' (1968) stemmer are shown in the next two columns. The SMART system (Salton, 1971) also proposes another English-language stemmer, and its evaluation is shown in the sixth column. Finally, the last column reports the retrieval performance obtained by applying a morphological analysis returning the lemma of each surface word.

The data depicted in Table 3 will be used to analyze two different questions. First, we will discuss the performance differences among IR models and explain with some detail the evaluation methodology in the rest of this section. Second, Table 3 will be used to compare the retrieval effectiveness of different stemmers together with the morphological analysis producing the corresponding lemma for each word. The latter will be discussed in the next section.

In Table 3 and in the following tables, the best performance under a given condition is depicted in boldface. Using this performance as a baseline, we then italicized those MAP values (in the same column) depicting statistically significant differences. Except for the second column, the DFR-I(n_e)C2 model always obtained the best results, statistically outperforming the classical *tf–idf* vector-space model or the LM scheme, from a statistical point of view. The same is usually true for a DFR-PL2 variant when compared to the best model. On the other hand, the MAP differences between Okapi and DFR-I(n_e)C2 are never statistically significant, implying that these two probabilistic models tended to perform at the same level.

When comparing two retrieval schemes, each overall statistical measure such as the MAP may hide performance irregularities among certain queries. For example, when comparing the DFR-I(n_e)C2 model with the classical *tf–idf* model, we found that the DFR-I(n_e)C2 model performed better for 245 queries, the classical *tf–idf* provided better AP for 27 queries, and both models maintained the same retrieval performances for the remaining 12 queries. To understand performance differences between these two models, we examined the largest difference obtained with Topic 62 ("Northern Japan Earthquake"). In this case, the DFR-I(n_e)C2 model obtained an AP of 1.0 while the classical *tf–idf* model obtained an AP of 0.0062. For this query, the *tf–idf* model's poor performance resulted from the fact that for some query terms, the term frequency was relatively high in the query. In this model, the documents at the higher ranks often contained one single search term with also a very high *tf* value (e.g., "Japan" with $tf = 125$ or $tf = 85$). Within the *tf–idf* model, this property ranked these documents higher compared to articles containing more query terms but having lower frequencies. For example, a relevant document having the search terms "earthquake" ($tf = 3$), "Japan" ($tf = 1$) and "report"($tf = 1$) was ranked in the first position with the DFR-I(n_e)C2, but was ranked in the 213th position with the *tf–idf* method.

Differences Between Stemming and Nonstemming Approaches

Table 3 also lists the results of following a verification of whether a stemmer's or a lemmatization application might improve retrieval performance when compared to a search strategy ignoring this type of word-normalization procedure. As shown in the second-to-last row of Table 3, we computed the average performance achieved by each of the five retrieval models to obtain an overview of the performance of each stemming approach. The last row shows the percent change when compared to an approach ignoring the stemming procedure. This value shows that the SMART stemmer obtained the highest average value of 0.4685 (or a relative improvement of +9.2% over the nonstemming method). The difference is rather small when comparing the SMART stemmer with other approaches such as that of Porter (0.4647) or when applying the morphological analysis (under the label "Lemma," 0.4597). In fact, for 159 queries, the S-stemmer improved retrieval performances while the nonstemming approach resulted in a better AP for the other 93 queries.

To verify whether these differences were statistically significant, we chose the performance labeled "None" as baseline. When using a stemmer or applying a lemmatization, if retrieval effectiveness was statistically significant, we placed

the symbol "†" after the corresponding MAP value. For example, when using the DFR-I(n_e)C2 IR model without stemming, we obtained a MAP of 0.4329 compared to 0.4658 when applying the S-stemmer. This difference was statistically significant, and was denoted by a "†" after the MAP value of 0.4658. Except for the Lovins' (1968) stemmer, all stemming approaches and the lemmatizer performed significantly better than did the nonstemming approach. The Lovins' stemmer tended to produce retrieval performances that were statistically similar to a nonstemming approach.

To obtain an overview of the precise effect of stemming, we analyzed concrete examples. With the DFR-I(n_e)C2 model, we saw that Topic 306 ("ETA Activities in France") retrieved a single relevant document and obtained an AP of 0.333 without stemming; after applying the S-stemmer, the AP was 1.0. The difference was due to the term "activities," which after stemming is reduced to "activity." The relevant document contains the term "activity" three times and "activities" two times. When conflated under the same stem, this search term was helpful in ranking the relevant article in first position after stemming.

Topic 98 ("Films by the Kaurismäkis"), on the other hand, retrieved only one single relevant document, with an AP of 1.0 before stemming and 0.5 after applying the S-stemmer. In this case, the single relevant document contains the term "films" nine times and the term "film" 12 times. After applying the S-stemmer, a nonrelevant document was ranked higher than the relevant article.

As another example, we could compare retrieval effectiveness using the DFR-I(n_e)C2 model and the SMART stemmer with the nonstemming approach (0.4329 vs. 0.4783). For Topic 180 ("Bankruptcy of Barings") using the SMART stemmer, the AP was 0.0082; without stemming, the AP was 0.7652. In this case, the word "Barings" was stemmed to "bare," which hurt retrieval performance. For Topic 63 ("Whale Reserve") using the stemmer, the AP was 1.0, meaning that the single relevant document was placed in the first position. Without stemming, the AP was only 0.0286, and the single relevant document was ranked 35th. Using the SMART stemmer, the word "Antarctic" occurring in the topic description was stemmed to "antarct," which would then match the word "Antarctica" appearing in the relevant document.

Similar findings can be obtained with other IR models such as the Okapi. For Topic 198 ("Honorary Oscar for Italian Directors"), returning a single relevant document obtains an AP of 0.5 without the stemmer and 1.0 with the SMART stemmer. Important changes in the query included the search terms "Honorary" (reduced to "honor") and "awarded" (stemmed to "award").

Algorithmic Stemmers or Morphological Analysis

As shown in the last line of Table 3, the percent of change obtained when comparing an approach ignoring the stemming procedure was rather similar across the different algorithmic stemmers or when applying a lemmatizer (under the label "Lemma"). To verify whether these differences are statistically significant, we selected the retrieval performance achieved with the SMART stemmer as a baseline. If the retrieval effectiveness was statistically significant, we indicated this by adding the symbol "‡" after the corresponding MAP value. For example, when using the DFR-I(n_e)C2 IR model with the SMART stemmer, we obtained a MAP of 0.4783 compared to 0.4565 when applying the Lovins' (1968) stemmer. This statistically significant difference was denoted by an "‡" after the MAP 0.4565. Performance differences also were significant for the other IR models, leading to the conclusion that the Lovins' stemmer results in lower performance levels than does the SMART stemmer.

During a query-by-query performance analysis comparing the Lovins' (1968) and the SMART stemmers, for Topic 98 ("Films by the Kaurismäkis"), the AP was 0.1429 with Lovins' stemmer and 1.0 for the SMART stemmer. The single relevant document was ranked in the seventh position with the Lovins' stemmer and in the first position by the SMART method. An analysis of the various stems produced by the two stemmers showed that with Lovins' method the stems were "ak" and "mik" whereas they were "aki" and "mika" with SMART stemmer. These two names came from the descriptive part of the topic formulation ("Search for information about films directed by either of the two brothers Aki and Mika Kaurismäki.") The stems produced by the Lovins' method were shorter and thus matched other unrelated terms in the rest of the collection.

On the other hand, Topic 231 ("New Portuguese Prime Minister") obtained an AP of 1.0 with the Lovins' stemmer and only 0.5 with the SMART stemmer. In this case, the single relevant item contained the noun "elections," which the Lovins' method reduced to the same stem as the adjective "electoral" appearing in the descriptive part of the topic. With the SMART stemmer, the noun and the adjective did not conflate under the same form, and the relevant item thus was not ranked first on the list.

The main conclusion, therefore, is that there are no statistically significant differences between efficient algorithmic stemmers such as Porter's, the SMART, or the S-stemmer and a morphological analysis returning the dictionary entry (or lemma) for each surface word. Thus, a light suffix-stripping algorithm such as the S-stemmer can achieve, *in mean*, a retrieval performance comparable to both the more aggressive algorithmic stemmers (Porter's, the SMART) or systems based on advanced natural language processing that correctly removes all inflexional suffixes.

Morphological Analysis, POS, and Thesaurus

In Table 4, we reported the MAP obtained using morphological analysis to produce the corresponding lemma for each inflected word form (same values as last column of Table 3). In the third column, we increased the document score when lemma common to the query and the retrieved item had the same POS tag. This feature could be useful in determining the precise meaning attached to a form. In the English language, the same term may have different meanings, depending on

TABLE 4. Mean average precision (MAP) for various IR models and different morphological analysis variants (284 title/description queries).

	MAP			
	Lemma	Lemma & POS	Lemma & Synset	Lemma, POS, & Synset
Okapi	0.4663	0.4720†	*0.4395*†	*0.4482*†
DFR-PL2	0.4608	*0.4634*	*0.4365*†	*0.4433*†
DFR-I(n_e)C2	**0.4671**	**0.4740**†	**0.4665**	**0.4705**
LM	*0.4444*	*0.4562*†	*0.4342*†	*0.4458*
tf–idf	*0.2778*	*0.2879*†	*0.2834*	*0.2888*†
Average	0.4597	0.4664	0.4442	0.4520
%change		+1.5	−3.4	−1.7

TABLE 5. Mean average precision (MAP) for various stop word lists using the S-stemmer (284 title/description queries).

	MAP		
	SMART	None	Short
Okapi	0.4648	*0.3403*†	0.4581
DFR-PL2	*0.4553*	*0.3185*†	*0.4526*
DFR-I(n_e)C2	**0.4658**	**0.4661**	**0.4665**
LM	*0.4493*	*0.4433*	*0.4462*
tf–idf	*0.2811*	*0.2831*	*0.2830*
Average	0.4588	0.3921	0.4559
%change		−14.5	−0.6

its POS, such as "lean" as an adjective (thin, lacking charm) or a verb (to recline or bend). The word "face" (or "form," "bank," "stem") may have a different meaning as a noun (a happy face) or as a verb (to deal with). To do so, for each indexing term a string composed of the term and its POS tag [e.g., with the adjective "alien," we added "alienJJ" in which "JJ" is the POS tag for the adjectives (Marcus et al., 1993)].

In the fourth column, we listed retrieval performances achieved by increasing the document score when query and documents had the same synset numbers. To do so, we added all synset numbers attached to an article or a query to its corresponding surrogate. Finally, in the last column of Table 4, we combined the two previous enhancements, which in turn assigned more weight when the terms common to both the retrieved records and the query also were the same POS and shared the synset numbers.

The results depicted in Table 4 confirm the conclusions we had drawn regarding the data shown in Table 3. The best performing IR model was still the DFR-I(n_e)C2, and the performance differences were always statistically significant (MAP italicized in Table 4) with both the LM or *tf–idf* models.

Compared to the morphological analysis only (performance under the label "Lemma" in Table 4 used as baseline), we might use the POS information to partly remove the ambiguity attached to search keywords. This additional information slightly improves the MAP, and the performance differences are always significant (MAP followed by the symbol "†" in Table 4), except for the DFR-PL2 model. For example, with the DFR-I(n_e)C2, the POS data increased the AP for 138 queries and decreased it for 98 (For the remaining 48, we obtained the same performance.) Using this IR model and Topic 217 ("AIDS in Africa"), the AP was 0.1944 under "Lemma;" yet, when we added the POS information, the AP increased to 0.5526. When inspecting the corresponding query, we first found that the stemming converted "AIDS" into "aid," and this increased the possibility of matches. When accounting for the POS tag, the stem "aid" was tagged as a proper noun, and thus improved the ranking of articles containing this abbreviation compared to documents containing either the singular noun "aid" or the plural form "aids."

Adding the thesaurus numbers to document and query representations (retrieval performance listed under the label "Lemma & Synset") tended to slightly decrease the MAP. For the three IR models, the differences were even statistically significant. With the Okapi model, for example, Topic 76 ("Solar Energy") obtained an AP of 0.663 under the "Lemma" condition, but only obtained an AP of 0.0722 under "Lemma & Synset." In this case, the description part of the topic contained the form "is" and "being" twice. The corresponding lemma "be" belongs to the 10 synsets added in the query surrogate (with a frequency of three). For each document containing a verbal form related to the verb "to be," we will thus have 10 query matches through the synset numbers, thus rendering discrimination between relevant and nonrelevant items more difficult.

Stop Word Lists

Finally, we have compared the retrieval effectiveness of the various IR models using different stop word lists. These lists contain words serving no purpose for retrieval purposes, but very frequently found in the documents. Upon removing these terms, each match between a query and a document would thus be based on relevant indexing terms. In other words, retrieving a document because it contains words such as "the," "has," "in," or "your" in the corresponding request does not constitute an intelligent search strategy. These nonsignificant words represent noise, and may actually damage retrieval performance because they do not discriminate between relevant and nonrelevant documents. Hopefully, we also would reduce the inverted file's size by from 30 to 50%.

In the second column of Table 5, we reported the retrieval performance achieved using the S-stemmer with the SMART stop word list containing 571 entries. This list may be viewed as relatively large, but Fox (1990) also suggested a relatively long list with 421 words. Next, we used the same stemming approach, but without any stop word list. The performance differences were small (∼1%) for the last three retrieval models when compared to the SMART stop word list, but relatively large for the Okapi (a relative decrease of −26.8%) and the DFR-PL2 (−30.1%) approaches. In the last column of Table 5, we used the short stop word list composed of nine words ("an," "and," "by," "for," "from," "of," "the," "to,"

"with") found in the DIALOG search engine (Harter, 1986). The average difference with the SMART stop word list is rather small (-0.6%), tending to indicate that the important point is to ignore only a short number of very frequent terms without any important meanings.

When applying the statistical tests (Significant differences are in italics while best performances are boldface.), we still can conclude that the DFR-I(n_e)C2 model is the best performing. When using the retrieval performance with the SMART stop word list as the baseline (second column), we found two cases in which the performance differences are statistically significant (MAP value followed by the symbol "†"). For example, when using the Okapi model, the MAP using the SMART stop word list is 0.4648, yet it decreases significantly to 0.3403 when accounting for all frequently occurring words (performances listed under the label "None"). Clearly, the performance achieved by either the Okapi or the DFR-PL2 is sensitive to the presence of very frequent words.

Through analyzing an example, we discover the main reasons for this phenomenon. Based on the Okapi model and applying the SMART stop word list, we obtained better retrieval performances for 223 queries while indexing all terms produced better AP for 37 queries (For the remaining 24 queries, the same AP was produced.) From an analysis of the extreme cases, we saw that Topic 136 ("Leaning Tower of Pisa") obtained an AP of 1.0 with the SMART stop word list, yet the AP was 0.0 when we accounted for all word forms. In the underlying query, the presence of many stop words (e.g., "of," "the," "is," "what") ranked many nonrelevant documents higher than the single relevant document.

On the other hand, with Topic 104 ("Super G Gold medal"), we obtained an AP of 0.6550 when ignoring the stop word list and an AP of 0.4525 with a stop word list. In this case, the search term "G" included in the stop word list was removed during the query processing. After this stop word removal, the final query was more ambiguous ("super gold medal") and could not rank the articles higher on the result list.

Conclusion

It has been recognized that the stemming procedure is an important component in modern IR systems and that an inappropriate stemmer may generate unexpected results to be presented to the user (Buckley, 2007; Savoy, 2007). Contrary to previous evaluations based only on the classical *tf–idf* vector-space model, we have shown that the same problem occurs with modern probabilistic models (e.g., Okapi, LM, or DFR), which perform significantly better than did the *tf–idf* approach.

Using a large set of queries ($n = 284$) extracted from the CLEF test collections, we also demonstrated that some algorithmic stemmers or a morphological analysis tend, *in mean*, to result in similar retrieval performances, at least for the English language. For medium-sized queries, the enhancement is around 7% greater than a search technique without stemming. For a language having a rather simple inflectional structure, this mean improvement is relatively high as compared to other languages. Using similar test collections (i.e., newspapers articles and comparable queries), Tomlinson (2004) obtained the following average improvements after stemming: +4% for Dutch, +7% Spanish, +9% French, +15% Italian, +19% German, +29% Swedish, and +40% Finnish. Recently, Kettunen (2007) also showed that when faced with languages with a complex morphology (e.g., Finnish or German), a morphological analysis tends to produce better retrieval performance than do algorithmic stemmers.

Among the various stemming approaches suggested for the English language, we found that the SMART (Salton, 1971), the Porter (1980), and the S-stemmer (Harman, 1991) methods as well as morphological analyses returning the corresponding lemma resulted in similar performance levels. Retrieval performance for the latter is significantly better than a nonstemming approach or the Lovins' stemmer (1968). In our opinion, this latter method removes too many final letters and thus is too aggressive, resulting in relatively short stems having high document frequencies. The examples presented earlier demonstrate some of these aspects.

When comparing stemming procedures, in our opinion, it is important to consider the final user. A nonstemming or a light stemming approach is better understood than is a more aggressive approach returning unexpected results. For this reason, in the English language we suggest using the S-stemmer (Harman, 1991), which only removes the plural form associated with English nouns.

We also tried to improve retrieval effectiveness by considering POS information and thesaurus class numbers. Compared to the morphological stemmer, accounting for the POS information will significantly improve MAP; however, the presence of the synset (or thesaurus class) numbers does not significantly modify mean retrieval performance, at least as implemented in this article.

Finally, it must be recognized that stop word lists were developed on the basis of certain arbitrary decisions (Fox, 1990). This is the case, for example, in commercial information systems, which tend to adopt a very conservative approach involving only a few stop words. According to our evaluations, the presence of a short stop word list containing around 10 terms produces retrieval effectiveness similar to that of longer stop word lists with 571 terms. Thus, not removing these very frequent terms with no real meaning may significantly hurt retrieval performance for some IR models (e.g., Okapi and DFR-PL2 in our experiments), when compared even to short stop word lists.

Acknowledgment

This research was supported in part by the Swiss NSF under Grant 200021–113273.

References

Amati, G., & van Rijsbergen, C.J. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Transactions on Information Systems, 20, 357–389.

Buckley, C. (2007). Why current IR engines fail. In Proceedings of ACM-SIGIR'2004 (pp. 584–585). New York: ACM Press.

Buckley, C., & Voorhees, E.M. (2005). Retrieval system evaluation. In E.M. Voorhees & D.K. Harman (Eds.), TREC: Experiment and Evaluation in Information Retrieval (pp. 53–75). Cambridge, MA: MIT Press.

Chan, Y.S., Ng, H.T., & Zhong, Z. (2007). NUS-PT: Exploiting parallel texts for word sense disambiguation in the English all-words tasks. In Proceedings of the 4th International Workshop on Semantic Evaluations (pp. 253–256), Stroudsburg, PA: ACL.

Fellbaum, C. (1998). WordNet: An electronic lexical database. Cambridge, MA: MIT Press.

Fox, C. (1990). A stop list for general text. ACM-SIGIR Forum, 24, 19–35.

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

Harter, S.P. (1986). Online information retrieval. Concepts, principles, and techniques. San Diego, CA: Academic Press.

Hiemstra, D. (2000). Using language models for information retrieval. Unpublished doctoral dissertation, Centre for Telematics and Information Technology, University of Twente, The Netherlands.

Hiemstra, D. (2002). Term-specific smoothing for the language modeling approach to information retrieval: The importance of a query term. In Proceedings of the ACM-SIGIR (pp. 35–41). New York: ACM Press.

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

Kettunen, K. (2007). Reductive and generative approaches to morphological variation of keywords in monolingual information retrieval. Unpublished doctoral dissertation, University of Tampere, Finland.

Krovetz, R. (1993). Viewing morphology as an inference process. In Proceedings of the ACM-SIGIR'93 (pp. 191–202). New York: ACM Press.

Lovins, J.B. (1968). Development of a stemming algorithm. Mechanical Translation and Computational Linguistics, 11, 22–31.

Manning, C.D., Raghavan, P., & Schütze, H. (2008). Introduction to information retrieval. Cambridge, England: Cambridge University Press.

Marcus, M.P., Santorini, B., & Marcinkiewicz, M.A. (1993). Building a large annotated corpus of English: The Penn Treebank. Computational Linguistics, 19, 313–330.

McNamee, P., & Mayfield, J. (2004). Character N-gram tokenization for European language text retrieval. Information Retrieval Journal, 7, 73–97.

Peng, F., Ahmed, N., Li, X., & Lu, Y. (2007). Context sensitive stemming for web search. In Proceedings of the ACM-SIGIR (pp. 639–646). New York: ACM Press.

Peters, C., Jijkoun, V., Mandl, T., Müller, H., Oard, D.W., Peñas, A., & Santos, D. (Eds.). (2008). Advances in multilingual and multimodal information retrieval. Lecture Notes in Computer Science (Vol. 5152). Berlin, Germany: Springer-Verlag.

Porter, M.F. (1980). An algorithm for suffix stripping. Program, 14, 130–137.

Ratnaparkhi, A. (1996). A maximum entropy part-of-speech tagger. In Proceedings of the Empirical Methods in Natural Language Processing Conference (pp. 133–142). Stroudsburg, PA: ACL.

Robertson, S.E., Walker, S., & Beaulieu, M. (2000). Experimentation as a way of life: Okapi at TREC. Information Processing & Management, 36, 95–108.

Salton, G. (1971). The SMART retrieval system—Experiments in automatic document processing. Englewood Cliffs, NJ: Prentice-Hall.

Savoy, J. (1993). Stemming of French words based on grammatical category. Journal of the American Society for Information Science, 44, 1–9.

Savoy, J. (1997). Statistical inference in retrieval effectiveness evaluation. Information Processing & Management, 33, 495–512.

Savoy J. (2007). Why do successful search systems fail for some topics? In Proceedings of the ACM Symposium in Applied Computing (pp. 872–877). New York: ACM Press.

Tomlinson, S. (2004). Lexical and algorithmic stemming compared for 9 European languages with Hummingbird SearchServer at CLEF 2003. In C. Peters, J. Gonzalo, M. Braschler, & M. Kluck (Eds.), Comparative evaluation of multilingual information access systems (pp. 286–300). Lecture Notes in Computer Science (Vol. 3237). Berlin, Germany: Springer-Verlag.

Voorhees, E.M. (1993). Using WordNet to disambiguate word senses for text retrieval. In Proceedings of the ACM-SIGIR'93 (pp. 171–180). New York: ACM Press.

Xu, J., & Croft, B. (1998). Corpus-based stemming using cooccurrence of word variants. ACM Transactions on Information Systems, 16, 61–81.

Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems, 22, 179–214.