This paper is being submitted to the Special Issue on Information Retrieval for Indian Languages of the ACM Transactions on Asian Language Information Processing (TALIP)

Sub-word Indexing and Blind Relevance Feedback for English, Bengali, Hindi, and Marathi IR

JOHANNES LEVELING
and
GARETH J. F. JONES
Centre for Next Generation Localisation
School of Computing
Dublin City University
Dublin 9, Ireland

The Forum for Information Retrieval Evaluation (FIRE) provides document collections, topics, and relevance assessments for information retrieval (IR) experiments on Indian languages. Several research questions are explored in this paper: 1. how to create create a simple, language-independent corpus-based stemmer, 2. how to identify sub-words and which types of sub-words are suitable as indexing units, and 3. how to apply blind relevance feedback on sub-words and how feedback term selection is affected by the type of the indexing unit. More than 140 IR experiments are conducted using the BM25 retrieval model on the topic titles and descriptions (TD) for the FIRE 2008 English, Bengali, Hindi, and Marathi document collections.

The major findings are: The corpus-based stemming approach is effective as a knowledge-light term conflation step and useful in case of few language-specific resources. For English, the corpus-based stemmer performs nearly as well as the Porter stemmer and significantly better than the baseline of indexing words when combined with query expansion. In combination with blind relevance feedback, it also performs significantly better than the baseline for Bengali and Marathi IR.

Sub-words such as consonant-vowel sequences and word prefixes can yield similar or better performance in comparison to word indexing. There is no best performing method for all languages. For English, indexing using the Porter stemmer performs best, for Bengali and Marathi, overlapping 3-grams obtain the best result, and for Hindi, 4-prefixes yield the highest MAP. However, in combination with blind relevance feedback using 10 documents and 20 terms, 6-prefixes for English and 4-prefixes for Bengali, Hindi, and Marathi IR yield the highest MAP.

Sub-word identification is a general case of decompounding. It results in one or more index terms for a single word form and increases the number of index terms but decreases their average length. The corresponding retrieval experiments show that relevance feedback on sub-words benefits from selecting a larger number of index terms in comparison with retrieval on word forms. Similarly, selecting the number of relevance feedback terms depending on the ratio of word vocabulary size to sub-word vocabulary size almost always slightly increases information retrieval effectiveness compared to using a fixed number of terms for different languages.

Categories and Subject Descriptors: H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance feedback, Query formulation*; H.3 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing methods, Linguistic processing*

General Terms: Experimentation, Measurement, Performance

Additional Key Words and Phrases: Information Retrieval, Evaluation, FIRE, Sub-word Indexing, Stemming, Blind Relevance Feedack

1. INTRODUCTION

Indian languages have a more complex morphology compared to English. Thus, information retrieval (IR) for Indian languages could profit from a more detailed analysis of stemming and sub-word extraction. In addition, blind relevance feedback (BRF) using sub-words has not been extensively discussed, yet. This paper provides a comprehensive investigation of stemming, sub-word indexing, and BRF. Their effectiveness is evaluated by conducting monolingual information retrieval (IR) experiments for the Indian languages Bengali, Hindi, and Marathi on the FIRE 2008 ad hoc test collections. For comparison, English monolingual IR experiments are also performed.

Stemming is a means to conflate different word forms to the same index term and has become a standard method to increase accuracy in IR. However, non-English IR and natural language processing (NLP) suffers from a lack of resources and tools in quantity and quality. This is particularly the case for languages which have not been the focus of significant previous IR evaluation campaigns such as the languages of the Indian subcontinent. To explore stemming for languages without existing resources, a language-independent corpus-based stemmer was created by analyzing suffix and root frequencies. The stemming procedure was inspired by an algorithm for morpheme induction developed for a morphological analysis of English and Bengali documents [Dasgupta and Ng 2007]. The goal of our work was to establish a simple but effective experimental baseline against which other retrieval experiments on the FIRE data can be compared. In contrast, stemming approaches for other languages typically build on additional languages resources which are not available or portable to other languages and often require additional processing, e.g. dictionary extraction, document conversions, or multiple scans over the document collection.

Stemming can also be seen as a special case of sub-word identification. A sub-word is the result of rewriting or removing parts of a word. Depending on the sub-word identification method, there may be more than one sub-word corresponding to the original word. Advantages of sub-word indexing over word or stem indexing include its robustness against spelling errors and orthographic variants and recall enhancement. Therefore, sub-word indexing has mostly been investigated for noisy data such as speech transcripts and in form of decompounding for compound-rich European languages such as German. Sub-word indexing may also prove useful for Indian languages with compounding properties (e.g. Marathi) or for IR on Indian newspaper articles in general, where the written representation of a word may differ from region to region. Three different sub-word indexing methods are investigated and compared to IR on word forms and stems: word prefixes, overlapping word-internal character n-grams, and sequences of vowels and consonants. The sub-word identification methods used here for the IR experiments described require little or no morphological analysis and also require no additional external resources such as dictionaries.

The main contribution of this paper is the investigation of blind relevance feedback in combination with sub-word indexing. To the best of the authors' knowledge, a relation between sub-word indexing and the number of relevance feedback terms and differences in the optimum number of feedback terms between different Indian

languages have not been investigated previously. Blind relevance feedback (BRF, also called pseudo-relevance feedback) builds on the assumption that the top-ranked documents retrieved for a query contain information which can be employed to reformulate a query (e.g. expand the query with additional terms) and to re-weight query terms. If – in comparison to word stems – more but smaller indexing units such as sub-words are indexed[1], the number of terms used in BRF should also be expected to be higher in comparison with BRF on indexed stems if a similar number of features associated with assumed relevant documents are to be included in the expanded query. This assumption for BRF is investigated for the different sub-word indexing techniques and all languages in the FIRE document collection.

In summary, corpus-based stemming and different sub-word indexing methods for robust retrieval on Indian languages are evaluated and compared. Finally, it is demonstrated that using smaller and more indexing units (compared to indexing words) requires the use of more feedback terms.

The rest of this paper is organized as follows: Sections 2 presents research and related work on stemming, sub-word indexing, and BRF, respectively. Section 3 describes the FIRE 2008 collection and processing steps for documents and topics. Section 4 introduces the new methods used for stemming, sub-word identification, and BRF. Section 5 contains a description of the experimental setup for the retrieval experiments and discusses results. The paper concludes with a summary and an outlook in Section 6.

2. RELATED WORK

Related research on stemming, sub-word identification, and blind relevance feedback is introduced in the next subsections. Selected official experiments on the FIRE 2008 data are briefly discussed in the final subsection of this section.

2.1 Stemming

The goal of stemming (affix removal) is to conflate different derivational or inflectional variants of the same word to a single indexing form. Stemming approaches can be classified into different categories, e.g. by the results produced by the stemmer (light stemming [Savoy 1999] vs. aggressive stemming [Lovins 1968]) or by the resources used (corpus-based [Xu and Croft 1998] vs. dictionary-based [Krovetz 1993; Majumder et al. 2007]).

Affix removal is language-dependent and creating a new stemmer involves analyzing text resources to produce a representation of morphological word formation rules. Hence, stemmers have to be customized for new languages, incorporating the language-specific morphological rules to form words. This can be an expensive and time-consuming task if language-specific resources are scarce, native speakers of the language are not available, or texts have to be analyzed manually. For example, dictionaries for Bengali, Hindi, and Marathi may be too small to be of practical use or they require additional processing if their contents are encoded in a Romanized transliteration but the document collection is not.

[1]In this context, *more* means multiple indexing units per word (and thus, more per document) and *smaller* means indexing units consisting of fewer characters than the original word.

Furthermore, adapting rule-based stemming manually or by rule translation to languages with a rich morphology might be too simplistic. For example, the English noun *"mouse"* and *"mice"* and the verb forms *"wear"*, *"wore"*, *"worn"* illustrate the so-called vowel mutation, which is a frequent effect in languages such as Arabic. These cases of word formation may be difficult to identify with automatic approaches.

Still, most stemmers are rule-based and are widely available only for English and other west European languages. There are three major stemmers in use for English IR: the Porter stemmer, the Lovins stemmer and the Krovetz stemmer.

The Porter stemmer [Porter 1980] employs rules which are successively applied if contextual prerequisites are met. Typical prerequisites include the number of consonant-vowel-consonant sequences (CVC) and character context before the suffix to be removed. The successive removal of affixes means that words with a recursive morphological structure are reduced to their base form, e.g. words such as *"hopelessness"* can be reduced to *"hope"* by removing all suffixes.

Lovins [1968] employs the most aggressive base form reduction strategy. The Lovins stemmer identifies word suffixes by a longest match strategy and applies more than 290 suffix removal rules, noting many exceptions. The aggressiveness of a stemming method is determined by the number of conflations it produces and is implied by the number of stemming rules or by the size and number of word clusters reduced to the same root form.

Krovetz [1993] introduced a stemmer which is known as Kstem. This stemmer is dictionary-based and requires that potential root forms should be contained in the dictionary, too. Experiments with this approach show a small, but significant increase in retrieval performance compared to indexing word forms.

Xu and Croft [1998] use a combination of aggressive suffix removal with co-occurrence information from small text windows to identify stemming classes. This technique is corpus-based and requires little knowledge about the document language. The original stemmer was developed for a Spanish document collection for which Xu and Croft report an increase in recall, but the stemmer could also be applied to other domains, corpora or languages.

Goldsmith [2001] identified suffixes employing a minimum description length (MDL) approach. MDL reflects the heuristic that words should be split into a relatively common root part and a common suffix part. Every instance of a word (token) must be split at the same breakpoint, and the breakpoints are selected so that the number of bits for encoding documents is minimal.

Oard et al. [2001] apply the Linguistica tool by Goldsmith [2001] to create a statistical stemmer. Suffix frequencies are computed for a subset of 500,000 words in a document collection. The frequencies of suffixes up to a length of 4 were adjusted by subtracting the frequency of subsumed suffixes. Single-character suffixes were sorted by the ratio between their final position likelihood and their unconditional likelihood. Suffixes were sorted in decreasing order of frequency, choosing a cutoff value where the second derivate of the frequency vs. rank was maximized.

Simple statistical stemmers typically take account of only the most frequent suffixes (e.g. *"-s"*, *"-er"*) and remove only the most frequent and shortest composite suffixes (e.g. *"-ers"*). A composite suffix such as *"-lessness"* is typically not rec-

ognized; instead, only the smaller, last suffix part *"-ness"* will be removed. Light stemming focuses on removing only the most frequent suffixes from word forms which are typically very few in number [Savoy 2006]. Recently, light stemming has been researched as a less aggressive means to reduce words to their root form. A well-known example for English is the *s*-stemmer, which removes only the *"-s"*, *"-es"*, and *"-ies"* suffixes from words (see, for example, Harman [1991]).

YASS is a clustering-based suffix stripper which has been applied to languages such as English, French, and Bengali [Majumder et al. 2007]. YASS identifies clusters of equivalence classes for words by calculating distance measures between strings. The stemmer does not handle morphologic prefixes and relies on several dictionaries which have to be extracted from a text resource, i.e. all words starting with the same character are collected in the same word list during preprocessing of the document collection.

In conclusion, many stemming approaches are data-driven, i.e. their development involves or is based on an analysis of collections of words from a dictionary or from the document collection. However, if a dictionary is already present certain rules to obtain the lexical base form for the dictionary must already exist.

The corpus-based stemming approach developed for the experiments described in this paper is considered to be language-independent, although there are some parameters which would require language-specific setting for best results. It requires no additional dictionary, handles composite suffixes, and can be extended to identify morphological prefixes as well. Stemming serves to provide experimental results which can be better compared to sub-word indexing.

2.2 Sub-word identification and indexing

The main idea behind sub-word indexing is to map word forms to one or more index terms. One objective of sub-word indexing is to overcome problems in IR which are caused by spelling and orthographic variants, morphological variants of the same word, or compound words. For example, the word form *"political"* may be represented by the sub-words *"poli"*, *"olit"*, *"liti"*, *"itic"*, *"tica"*, *"ical"*. Related words such as *"politician"* or *"politics"* share some of the sub-words, which allows for partial matches.

Decompounding a word into its constituent words is a special case of sub-word identification. For languages with a rich morphology (such as Finnish, Dutch or German), a linguistically motivated decomposition of words has been widely recognized as a method to improve IR performance [Hedlund 2002; Braschler and Ripplinger 2003; Chen and Gey 2004]. In languages such as English, compounds are typically written as separate words and their constituents can be easily identified. For languages such as German or Marathi, compounds are written as single words and IR may benefit from linguistically motivated decompounding.

Glavitsch and Schäuble [1992] and Schäuble and Glavitsch [1994] extract consonant-vowel-consonant (CVC) sequences as indexing features for retrieval of speech documents to obtain a more robust approach for noisy speech transcriptions. They select features based on document and collection frequency, and discrimination value. This indexing method performed slightly better than one using stopword removal and stemming. Similarly, Ng [2000] performs experiments with CVC on spoken documents for English, achieving a 28% performance increase when com-

bining sub-words indexing with error compensation routines. For Indian language texts, characters may alternatively be represented by different glyphs which appear visually similar (e.g. the letter O may be represented by a sequence of the letter A and the sign O or by the letter A, followed by signs for E and AA), which poses an indexing problem similar to that caused by errors in optical character recognition of printed materials.

McNamee [2001] performs retrieval experiments using overlapping character n-grams as indexing units. He reports performance results for indexing a combination of 2-grams, 3-grams, and 4-grams for English, Chinese, Japanese, and Korean. Results show that n-grams can achieve similar or superior performance in comparison to standard indexing techniques (e.g. indexing words), even for non-compounding languages and for cross-lingual retrieval [McNamee and Mayfield 2007].

McNamee [2008] also performs BRF experiments for different sub-word identification techniques. He shows that these techniques have a different optimum number of feedback terms and states that 25 terms are optimal for word-based indexing and 200 terms are a good choice for n-grams in English and other languages. He also demonstrates that BRF on n-grams shows only 2-4% absolute improvement in MAP compared to 9% for BRF on indexed words. (Unless noted otherwise, performance improvements in this paper are reported relative to the baseline experiments.)

Braschler and Ripplinger [2003] give an overview of stemming and decompounding for German. They perform IR experiments on data from CLEF for the ad-hoc retrieval track. A variety of approaches for stemming and decompounding are applied, including commercial solutions, resulting in a performance gain of up to 60.4% mean average precision (MAP) and 30.3% for the number of relevant retrieved documents in comparison to indexing raw word forms (not stems).

Hedlund [2002] investigates compound splitting for cross-language IR using a dictionary-based approach. For experiments on German, Swedish, and Finnish based on CLEF data, it was found that compound processing (i.e. decompounding into sub-words) has in general a positive effect on retrieval performance.

In Asian languages with logographic script such as Chinese, text does not contain delimiters indicating word boundaries (e.g. whitespace between words). Hence, word segmentation and sub-word identification as a special case are important language processing tasks [Foo and Li 2004; Ogawa and Matsuda 1997; Chen et al. 1997]. Most Chinese words are character bigrams. On the Chinese TREC 5 collection, Chen et al. [Chen et al. 1997] found that dictionary-less bigram methods perform similarly or better than dictionary-based methods.

Other approaches to sub-word indexing include dictionary-based lemmatization [Leveling and Hartrumpf 2005], and determining syllable-like character sequences using Knuth's algorithm for hyphenation [Leveling 2009].

In summary, sub-word indexing has mostly been investigated for noisy data and for compound-rich European languages. In Indian texts, the same word can have multiple written variant forms, which means that texts may have some of the properties of transcribed speech or documents obtained via optical character recognition. This makes sub-word identification worth investigating for Indian languages.

2.3 Query Expansion by Blind Relevance Feedback

Blind relevance feedback techniques for query expansion have been widely used in information retrieval to improve performance. Typical IR experiments compare BRF with different numbers of documents or terms extracted, or vary the term selection criterion.

Buckley et al. [1994] perform massive query expansion with the SMART retrieval system for ad-hoc retrieval experiments at TREC 3. They employ Rocchio feedback [Rocchio 1971] with 300 to 530 terms and phrases for each topic. An improvement of retrieval effectiveness between 7% and 25% in various experiments was observed.

Sparck-Jones et al. [2000] vary the number of allowed feedback terms depending on the query length. They tried to identify the optimum number of feedback terms depending on the query size. Supported by evidence from their results, they suggest using 16, 24, and 32 relevance feedback terms for short, medium, and long queries, respectively.

Lynam et al. [2004] compare six retrieval systems capable of BRF with default system settings. They observed that the lowest performing method profits most from feedback. The system producing the initial result set with the highest performance, the SMART system, showed the least increase in retrieval effectiveness.

Billerbeck and Zobel [2004] question the usefulness of query expansion. They perform experiments using a simplified BM25 formula, appending 25 terms "with the smallest [sic!] TSVs" (term selection values) to the query and downgrading the Okapi term weight by 1/3. They found that query expansion is in some cases not effective, but failed to provide a more detailed explanation as to why this is the case.

There are many other query expansion strategies including EVM [Gey et al. 2001], global analysis [Xu and Croft 1996], and implicit feedback [Shen et al. 2005], but in this paper we focus on studies directly relevant to our retrieval experiments.

2.4 Experiments at FIRE 2008

The FIRE 2008 evaluation initiative has attracted various IR research teams. Table I shows results for the best performing experiments. McNamee [2008] employs n-grams and skipgrams as indexing units for IR on English, Bengali, Hindi, and Marathi documents using language modeling (LM) as a retrieval model. Skipgrams are n-grams with wildcards [Pirkola et al. 2002], and have been investigated by Guthrie et al. [2006] to overcome the data sparsity in n-gram modelling. McNamee experimented with different but fixed numbers of expansion terms for different indexing methods: 50 feedback terms for words, 150 for 4-grams and 5-grams, and 400 for skip-grams.

McNamee et al. [2009] conduct additional IR experiments on the FIRE 2008 data. They compute n-grams on running text, treating whitespace as part of the n-grams. They investigate different term indexing methods, including word-spanning n-grams, truncation, and word-internal n-grams. Significant improvements for indexing sub-words compared to the baseline of indexing words are observed. The best effectiveness for Hindi and Bengali is achieved by 4-grams, for Marathi by word-internal 4-grams.

Paik and Parui [2008] use the Terrier system on the FIRE data. Their experi-

Table I. Top performing IR experiments for the FIRE 2008 document collections.

Language	MAP	Method/Group
English	0.5572	Language modeling for IR [Udupa et al. 2008]
Bengali	0.4719	Combination of methods (DFR, LM, Okapi) on sub-words and data fusion (on TDN) [Dolamic and Savoy 2008]
Hindi	0.3487	5-gram indexing [McNamee 2008]
Marathi	0.4575	Combination of methods (Okapi, DFR) on sub-words and data fusion (on TDN) [Dolamic and Savoy 2008]

ments make use of the title, description, and narrative field of topics (TDN). They redefine characters based on their context as dependent or compound characters. For stemming, they reduce word forms to their common prefixes in a single scan over a lexicon. The best performance achieved in their experiments was 0.4232 MAP for Bengali, 0.2709 MAP for Hindi, and 0.4239 MAP for Marathi, respectively.

Dolamic and Savoy [2008] use the Okapi IR model, divergence from randomness (DFR) and language modeling (LM) for FIRE 2008 experiments on Bengali, Marathi, and Hindi documents. Their approach includes light stemming [Savoy 2006] and stopword removal based on small stopword lists (less than 200 words). They also apply Rocchio feedback (with $\alpha = \beta = 0.75$) using 3-10 feedback documents and 20-100 feedback terms and find that blind relevance feedback seems to be a useful techniques for enhancing retrieval effectiveness. The best performance is based on data fusion of results from different IR models. For Bengali, the best result was 0.4719 MAP using the TDN fields from the topics.

Xu and Oard [2008] apply a Perl Search engine on the FIRE data for English-Hindi CLIR. In preliminary experiments on the FIRE data, they employ a stopword list with 275 words for Hindi IR, using BM25 with default parameters ($b = 1.2$, $k1 = 0.75$, $k3 = 7$). The monolingual Hindi baseline in these experiments is 0.37 MAP, which was improved to 0.38 MAP when using query expansion.

In summary, different languages require different indexing and retrieval approaches. No best practice for IR on Indian languages has been established, yet, but there seems to be a trend towards simpler approaches (using fewer stopwords, light stemming, and knowledge-light processing). FIRE is the first attempt to provide test collections to form a retrieval benchmark for Indian language IR.

3. INDIAN LANGUAGE IR BASED ON FIRE DATA

The Forum for Information Retrieval Evaluation (FIRE) provides large-scale document collections for Indian language information retrieval experiments. Similar to other IR evaluation initiatives such as TREC[2], NTCIR[3], or CLEF[4], FIRE aims at comparing the retrieval performance of different systems and approaches and at investigating evaluation methods for IR. FIRE started in 2008 with document collections for English, Bengali, Hindi, and Marathi.

[2] http://trec.nist.gov/
[3] http://research.nii.ac.jp/ntcir/
[4] http://www.clef-campaign.org/

```
<DOC>
<DOCNO> 1060312_foreign_story_5958714.utf8 </DOCNO>
<TEXT> The Telegraph - Calcutta : International Police end Paris student protests

Students (right) confront policemen outside the Sorbonne University, Paris. (AFP)

Paris, March 11 (Reuters):
French riot police used teargas to break up a three-day sit-in at Pariss Sorbonne
university today, stirring up memories of May 1968, as angry students warned of
a mounting challenge to government labour reforms.
... </TEXT>
</DOC>
```

Fig. 1. Sample English FIRE document.

```
<top lang="en">
<num> 41 </num>
<title> New Labour Laws in France </title>
<desc> Find documents reporting on the introduction of the new labour law in
 France and the protests against it. </desc>
<narr> Information regarding the introduction and adoption of the First
 Employment Contract in France by the Parliament, the giant protests by the
 unions, students and youth in response to the new employment legislation, the
 assault on the students by the French riot police, the announcement of the
 French President to scrap the CPE and the threat by the union leaders that
 the government would face a repeat of the recent general strikes if the law
 were not withdrawn. </narr>
</top>
```

Fig. 2. Sample English FIRE topic.

3.1 Documents and Topics

The FIRE document collection contains newspaper articles on various topics including sports, politics, business, and local news. The articles are represented as structured XML documents in TREC format, using UTF-8 encoding. Figure 1 shows an excerpt from a FIRE document.

FIRE topics resemble those from other retrieval campaigns such as TREC in format and content. They comprise a brief phrase describing the information need (topic title, T), a longer description (topic description, D), and a part with information on how documents are to be assessed for relevance (topic narrative, N). Retrieval queries are typically generated from the title and description fields of topics (TD). Figure 2 shows a sample FIRE topic. For each language, fifty unique topics and the relevance assessments were provided together with the corresponding document collection. For all FIRE topics, relevant documents have been assessed by pooling submissions from systems participating in the FIRE retrieval track.

Statistics about the FIRE document collections are shown in Table II (Avg_doclen: average document length in terms). Every document was provided as a single file. Not all documents could be indexed properly: some files include invalid XML characters or contain otherwise invalid XML; others contain no valid text at all. These

Table II. Statistics for the FIRE 2008 document collections.

Language	# Files	# Docs (%)	Size (MB)	Avg_doclen
English	125,638	125,583 (99.9)	580	268.2
Bengali	123,047	123,044 (99.9)	966	265.3
Hindi	95,215	94,963 (99.7)	926	554.7
Marathi	99,926	99,357 (99.4)	684	270.9

Table III. Statistics for the FIRE 2008 topics.

Language	# Topics	# Assessed	# Relevant	Avg_relevant
English	50	18,656	3,779	75.59
Bengali	50	11,967	1,863	37.26
Hindi	50	23,587	3,436	68.72
Marathi	50	8,155	1,095	21.90

documents have not been indexed at all, but they make up only a small portion of each collection. In addition, some duplicate document identifiers (IDs) were identified in the document collections. Duplicate document IDs were discarded from the output of the retrieval system used for the experiments described in this paper, i.e. document IDs occuring more than once were omitted. Table III shows statistics of the topics for FIRE 2008 (# Assessed: number of documents assessed for relevance; # Relevant: number of relevant documents; Avg_relevant: average number of relevant documents per topic).

3.2 Language-specific Preprocessing

The stopword lists for English, Bengali, Hindi, and Marathi used for the experiments described in this paper originate from different sources. The first source of stopwords is Jacques Savoy's web page on multilingual resources for IR at the University of Neuchâtel[5], which contains links to files with stopwords in many languages. These stopword lists have been generated by the Neuchâtel group following an approach to obtain a general stopword list for a document collection [Fox 1992; Savoy 1999], in which the N most frequent words are extracted from a document collection, numbers are removed from the list, and the resulting stopword list has been manually extended with additional word forms. The stopword lists from this web page contain 571 words for English (the SMART stopword list), 119 for Bengali, 163 for Hindi, and 98 for Marathi.

Second, special characters and punctuation marks were compiled by us in a list. For example, "।" is used as an end-of-sentence marker in Bengali. Finally, a stopword list is created during our indexing experiments, containing the most frequent index terms. Terms occurring in more than 75% of all documents in the document collection are considered as stopwords.

For the IR experiments on the FIRE document collections, some minimal knowledge of Indian languages is essential. For example, text processing in IR typically involves case normalization, and some sub-word indexing methods require differen-

[5] `http://members.unine.ch/jacques.savoy/clef/index.html`

tiating between vowels and consonants.

The English (Roman) writing system or script is based on 26 characters, of which six can be vowels. English uses a left-to-right writing system with capitalization of characters in the initial position of a sentence and capitalization of proper nouns. Several punctuation marks (e.g. *"?"*, *"!"*, *";"*) are used.

The Bengali writing system employs eleven graphemes denoting the independent form of nine vowels and two diphthongs. There are thirty-nine letters denoting consonants with so-called inherent vowels. Bengali has a left-to-right writing system and uses no capitalization. In addition to punctuation marks incorporated from Western scripts, Bengali script defines a symbol for the full stop, i.e. *"।"*.

Hindi and Marathi are written in Devanagari, a script consisting of 52 letters (16 vowels and 36 consonants). Devanagari is written from left to right and uses no capitalization. Marathi is a compounding language, i.e. two words can be joined together by a morphological process to form a new word (written as a single word or combining the two words with a hyphen).

The set of vowels differs from language to language: In English, vowels are *"a"*, *"e"*, *"i"*, *"o"*, *"u"* and *"y"* if preceded by a consonant. For the experiments described in this paper, vowels are also determined by their character context (e.g. *"y"*). Vowels in other European languages include letters with accents and diacritical marks such as the French letter *"é"* or the umlaut character *"ä"*. In Indian script, all consonants have an inherent vowel. A change to the inherent vowel is indicated by adding a vowel sign to the base consonant. The vowel sign can appear left, right, above, below or on both sides of the base consonant. For example, the vowel AA appears to the right and the vowel I appears to the left of a consonant in Devanagari. Vowels independent of a consonant appear at the beginning of a word or directly after a vowel. Thus, vowels in Indian script do not depend on character context and are encoded by different characters.

Special attention has been given to character normalization. Larkey et al. [2003] normalize Hindi multi-byte characters using manually crafted rules for the TIDES (Translingual Information Detection, Extraction, and Summarization) surprise language exercise. Unnormalized text encoded with UTF-8 may use different multi-byte character encodings for the same character. For example, the character *é* in the Spanish name *San José* may be encoded as a single byte (for *é*), as the byte sequence for *e* + ´ or as the byte sequence for ´ + *e*. For the experiments described in this paper, encoded text was normalized by following the guidelines for canonical decomposition followed by canonical composition from the International Components for Unicode (ICU) implementing the standard normalization forms, which is described in the Unicode Standard Annex #15 - Unicode Normalization Forms[6]. These normalization steps guarantee a fixed order of characters where multiple variants are allowed. The normalization of data for the IR experiments was motivated by the following reasons:

—Other researchers have reported inconsistencies or variations in encoding Indian documents [Larkey et al. 2003], news articles [Pal et al. 2006], and web pages [Pingali et al. 2006].

[6] `http://www.unicode.org/unicode/reports/tr15/`

—The FIRE 2008 documents originate from different sources and are written in different languages. Newspaper agencies publishing articles and newswires are located in different regions and may use different encoding guidelines (or do not use any guidelines at all), even for the same language.

—Proper nouns play an important role in IR. In Indian documents, foreign proper nouns may be transcribed or transliterated in different ways, similar to English variants of the name *Gorbachev* originating from transliteration (e.g. *Gorbatschow*).

—Additional or external resources (e.g. stopword lists or dictionaries) may use a different character encoding which would make resources incompatible.

Normalizing the FIRE documents affected about 13.6% of all tokens in Bengali, 2.7% in Hindi, and 6.1% in Marathi. The English documents were not changed at all by the normalization.

Incomplete or missing normalization of resources would result in many mismatches in preprocessing (e.g. unrecognized stopwords) and retrieval (e.g. spelling variants in documents). In addition, text was processed by applying the following normalization rules, which are inspired by Larkey et al. [2003] and Pingali et al. [2006].

(1) Internal word space is removed (e.g. characters U+200C and U+200D).

(2) *Chandrabindu* and *Anusvara* are special characters indicating nasalization sounds. Both *Chandrabindu* and *Anusvara* are mapped to *Anusvara* because they are mutually exclusive but phonetically similar.

(3) *Chandrabindu* followed by a vowel is mapped to the corresponding vowel.

(4) Consonants in Indian languages typically have an inherent vowel sound. The *Virama* diacritic, also called *Halanta*, is placed below a character to delete the vowel sound. *Virama* characters are removed from the text because they are often implied and optionally written.

(5) *Nukta* is a combining character used to obtain additional consonants with are encoded separately. It represents sounds in other languages such as English. Combinations of *Nukta* and a consonant are replaced by the corresponding consonant character.

(6) Long vowels are mapped to the corresponding short form, as has been suggested by Pingali et al. [2006] for word spelling normalization in web search.

(7) Some character sequences which are visually similar to a single glyph are mapped to a single character (e.g. letter A + sign O, letter A + sign AA + sign E, letter A + sign E + sign AA are mapped to the letter O). An example of visually similar character sequences is shown in Figure 3.

(8) Accents are typically part of transcribed foreign names. They are removed, because they may not be used consistently.

(9) Digit symbols in Bengali and Devanagari are mapped to Arabic numeric literals, because the FIRE data contains both forms.

letter A (U+0985) + sign O (U+09CB): অ + ে + া = ো

letter A (U+0985) + sign E (U+09C7) + sign AA (U+09BE): অ + ে + া = ো

Fig. 3. Example for visually similar character sequences.

Table IV. Top twenty English suffixes extracted by morpheme induction.

Suffix	Example	Suffix	Example
"-s"	play+s	*"-es"*	furnish+es
"-ing"	twang+ing	*"-e"*	retriev+e
"-ed"	disarm+ed	*"-pur"*	ekbal+pur (police station in Kolkata)
"-ly"	brave+ly	*"-r"*	write+r
"-a"	saheb+a (Indian Cricket player)	*"-n"*	bahrai+n (country)
"-ness"	bright+ness	*"-an"*	dominic+an
"-er"	barb+er	*"-y"*	hors+y
"-ers"	barb+ers (composite suffix)	*"-as"*	somak+as (tribe in ancient India)
"-i"	damir+i (Egyptian writer)	*"-rs"*	governo+rs
"-d"	lai+d	*"-ally"*	clinic+ally (composite suffix)

4. EXPERIMENT PREPARATIONS

4.1 Morpheme Induction for a Corpus-based Stemmer

Removing a fixed number of suffixes from words in different languages might result in a more or less aggressive stemming approach. The Porter stemmer for English applies about 50 rules roughly corresponding to a single simple suffix and is typically considered as moderately aggressive. Removing the same number of suffixes from words in a different language may result in very light stemming. For example, Bengali has a much richer morphology than English and has more complex word formation rules, which is indicated by the higher number of possible morphological suffixes.

A corpus-based, language-independent stemming approach was implemented following a morpheme induction approach which has been evaluated for English and Bengali and is described in [Dasgupta and Ng 2007] and [Dasgupta and Ng 2006]. On a manually annotated set of Bengali words this approach achieved a substantially higher F-score than Linguistica [Goldsmith 2001].

For the retrieval experiments described in this paper, the first steps of the morpheme induction were implemented to obtain a stemmer. The later morpheme induction steps described by Dasgupta and Ng [2007] mainly test the validity of composite suffix candidates and suffix attachments. The morpheme induction produces a list of candidate suffixes based on a frequency analysis of potential word roots and suffixes. For example, the word *"hopeful"* can be split into the root-suffix pairs *"hop"*+*"eful"*, *"hope"*+*"ful"*, and *"hopef"*+*"ul"*. The middle variant is chosen, because its root and suffix frequency are highest. In a second step, suffix combinations (composite suffixes) are determined via the frequency of potential root forms, allowing for a recursive morphological word structure. A word is stemmed by removing the longest suffix found in the generated suffix lists or by not removing

a suffix, otherwise.

The list of candidate suffixes is produced using a method suggested by Keshava and Pitler [2006]. The top twenty suffixes for English are shown in Table IV. For readability and for a better comparison to suffixes removed by other stemming approaches, the examples are given in English. Note that all of the suffixes recognized by the *s*-stemmer are included in the lists generated by the morpheme induction. The top suffixes also contain some composite suffixes (*"-ers"* and *"-ally"*), which were identified as simple suffixes in the first step of morpheme induction. Possible improvements of the stemmer include calculating the updated frequencies of suffixes (as suggested by Oard et al. [2001]), and removing proper nouns from the document collections before morpheme induction. While the examples given in Table IV were randomly selected from the set of root candidates, some suffixes seem to be representative for proper nouns only. For example, the suffix *"-a"* was identified as a probable suffix from the FIRE document collection, probably because the proper noun *"India"* is present in many newspaper articles. Goldsmith [2001] lists some erroneous cases where proper nouns are incorrectly stemmed and assigned to word sets of the same morphological signature. However, no previous work has proposed that removing proper nouns from the training data or document collection might improve the accuracy of affix removal. As an obvious improvement of the morpheme induction, we suggest to apply named entity recognition to the document collection and exclude named entities from the training data. While named entities follow morphological rules like other words, it may be safe to assume that they morphologically behave like other nouns (but with different frequency patterns). Thus, identifying and removing proper nouns will likely improve the accuracy of stemming, because common word suffixes which are part of proper nouns are excluded. For example, first names and location names in newspaper articles may be specific to a region or culture (in this case India, see Table IV) and not specific to word formation rules of the document language (in this case English). However, the investigation of this approach is beyond the scope of this paper.

After indexing the word forms in a document collection (as is done for each language in the baseline experiments described here), the index contains all terms and their surface frequency which is extracted for morpheme induction. All words w are analyzed by successively selecting all possible segmentation points, splitting them into a potential root form r and a suffix s. Thus, w is the concatenation of r and s. The morpheme induction method is also applicable to determine linguistic prefixes, but the stemmer described in this paper only removes word suffixes. However, most stemmers remove suffixes only, because removing a prefix may also change the meaning of a word to its antonym (e.g. *"legal"* vs. *"illegal"*). It is presumed and it is usually the case that the collection vocabulary will not only contain forms corresponding to inflected or derived words, but also the uninflected root forms. If the potential root form r is contained in the set of index terms (e.g. it is part of the collection vocabulary and the root frequency is higher than 0), s is added to the list of suffix candidates and r is added to the list of root candidates. Candidate suffixes are filtered as follows:

(1) In a minor variation of the approach proposed in [Dasgupta and Ng 2007], suffixes with a frequency (i.e. the number of words to which this suffix is

attached) less than a given threshold t_f are removed (in this case, $t_f < 5$).

(2) A score is assigned to each suffix by multiplying the suffix frequency and the suffix length in characters. Using suffix length as a scoring factor is motivated by the observation that short, low-frequency suffixes are likely to be erroneous [Goldsmith 2001].

The suffix candidates are then ranked by their score to obtain the top K suffixes. Dasgupta and Ng [2007] state that the parameter K depends on the vocabulary size and use different values for English and Bengali (given a similar collection size for different languages). For the experiments described here, a fixed value of $K = 50$ was used for all languages tested. Dasgupta and Ng [2007] used the same number of suffixes for morpheme induction for English. Considering that about 50 affix removal rules are defined by the Porter this seems a plausible setting for mildly aggressive stemming.

Composite suffixes are detected by combining all suffixes in the induced candidate list, e.g. *"-less"+"ness"* in *"fearlessness"* where + denotes the concatenation of strings. For morphologically rich languages like Arabic or Bengali, composite suffix detection plays an important role in base form reduction. The detection of composite suffixes $s_1{+}s_2$ builds on the assumption that a root form r will also combine with part of the suffix (s_1). This property typically does not hold for non-composite suffixes. The morpheme induction method presumes that $s_1{+}s_2$ is a composite suffix if $s_1{+}s_2$ and s_1 are similar in terms of the words they can combine with. Specifically, $s_1{+}s_2$ and s_1 are considered to be similar if their similarity value – which is calculated as shown in Equation 1 – is greater than a threshold t_s (specifically, $t_s > 0.6$ was used).

$$similarity(s_i + s_j, s_i) = P(s_i|s_i + s_j) = \frac{|W^{iji}|}{|W^{ij}|} \tag{1}$$

where $|W^{iji}|$ is the number of distinct words that combine with both $s_i{+}s_j$ and s_i, and $|W^{ij}|$ is the number of distinct words that combine with $s_i{+}s_j$. These values correspond to the morphological family size of $s_i{+}s_j$ and its intersection with s_i, respectively.

The analysis of suffixes is performed after indexing words forms. All index terms are processed as described and the top ranked suffix candidates and composite suffix candidates are extracted. For all tested languages, the morpheme induction process takes less than a minute to finish (using a standard PC), which is much less time than for indexing the document collections. The corpus-based stemmer reads the lists of suffixes and processes words which are longer than a given threshold t_l ($t_l = 3$). All other words remain unstemmed. The stemmer determines the one longest suffix in the suffix lists (if any) and removes it from the word to produce a root form.

4.2 Sub-word Identification

Sub-word identification aims at breaking up long words into smaller units. These units are generated by methods such as decompounding words into lexical constituent words or by splitting words into character n-grams of a fixed size. Splitting

Table V. Examples for splitting the English words *"information retrieval"* into sub-words with different methods.

Method	Sub-word type	# Sub-words
stem	informat, retriev	2
4-prefix	info, retr	2
5-prefix	infor, retri	2
6-prefix	inform, retrie	2
3-gram	inf, nfo, for, orm, rma, mat, ati, tio, ion, ret, etr, tri, rie, iev, eva, val	16
4-gram	info, nfor, form, orma, rmat, mati, atio, tion, retr, etri, trie, riev, ieva, eval	14
5-gram	infor, nform, forma, ormat, rmati, matio, ation, retri, etrie, triev, rieva, ieval	12
CV	i, nfo, rma, tio, n, re, trie, va, l	9
VC	inf, orm, at, ion, r, etr, iev, al	8
CVC	inf, nform, rmat, tion, n, retr, triev, val, l	9
VCV	info, orma, atio, ion, r, etrie, ieva, al	8

a compound word and finding smaller indexing units will usually make a match more likely and yield a higher recall. Linguistically oriented approaches restrict sub-words to constituent words. Other approaches to create sub-words do not require that sub-words must be valid words of the language (e.g. character n-grams [McNamee et al. 2009]).

In addition to corpus-based stemming, three different methods of sub-word indexing are investigated and compared in this paper: n-prefixes, word-internal n-grams (short: n-grams), and CVC sequences. These methods were selected as they do not rely on extensive linguistic resources for Indian languages and they aim at providing robust performance for potentially noisy data. Table V shows results of applying sub-word splitting techniques to the English phrase *"information retrieval"*. The following subsections provide a more detailed description of these sub-word indexing techniques.

Word truncation is a method which can be easily applied to processing text for writing systems with a fixed writing order. Word forms are truncated after the n-th character.[7] McNamee et al. [2009] describe experiments using truncated words and n-grams in many languages, including Bengali, Hindi, and Marathi. In their experiments, word-spanning 5-grams outperform word-internal 5-grams for Bengali and Hindi. For Marathi, word-internal 5-grams perform better. For all languages, 5-prefixes perform slightly worse than word-spanning 5-grams. They conclude that word truncation (here called n-prefix) shows a significant increase in retrieval effectiveness and may be a viable alternative to using a stemmer for resource-scarce languages. For the experiments described in this paper, overlapping word-internal n-grams were employed.

Overlapping character n-grams (3-grams, 4-grams, and 5-grams) have been suc-

[7]In this paper, the term n-prefix is used to denote word truncation of a word after at most n characters. The distinction between n-prefixes and linguistic or morphologic prefixes of a word should be clear from the context.

cessfully used for monolingual and cross-lingual IR (see [McNamee et al. 2009; McNamee 2001; McNamee and Mayfield 2007; McNamee 2008]). Words can be broken up into sequences of characters of a fixed size n to form character n-grams. If n-grams are allowed to start at every character position (instead of one n-gram for every n characters), the n-grams will partially overlap. Some variants of this method add an extra character as a special word boundary marker to n-grams from the beginning and end of a word. Following this approach and using the character *"|"* as a boundary marker, the set of 5-grams for the noun *"membership"* includes the gram *"ship|"* from the ending of the word and allows us to distinguish it from n-grams for the noun *"shipping"*.

In another approach, the full text is regarded as a single string and not broken down into words before calculating n-grams. Whitespace characters are not distinguished and become part of the character n-grams, which can span word boundaries (word-spanning n-grams). For languages with a writing system that rarely uses whitespace, such as Chinese, this is the default behavior.

Identifying consonant-vowel sequences requires classifying characters into vowels and consonants. A CVC sequence is the longest match of a sequence of zero or more consonants (C), followed by zero or more vowels (V), followed by one or more consonants in a word. Three variants of these character sequences can be defined accordingly (VCV, CV, and VC sequences) and are investigated in this paper too. Consonant-vowel sequences (CV) and variant methods, including vowel-consonant sequences (VC), consonant-vowel-consonant sequences (CVC), and vowel-consonant-vowel sequences (VCV) were often used for noisy data, e.g. in speech retrieval [Glavitsch and Schäuble 1992].

4.3 Term Selection for Blind Relevance Feedback on Sub-words

Blind relevance feedback has been applied in many ad-hoc IR experiments. To the best of the authors' knowledge, the existence of a relationship between the size or number of indexing units and the preferred number of feedback terms has not yet been investigated.

Splitting words into sub-words will typically produce more but smaller indexing units (cf. Table V and Table VI). However, smaller indexing units can be more ambiguous, i.e. they occur more frequently and they may originate from different word forms and introduce ambiguity. The relationship between the number of feedback terms to be used and the type of the indexing unit has not been thoroughly investigated because most IR experiments focus on retrieval on indexed stems. Furthermore, BRF on indexed words may expand a query with morphologic or spelling variants of a query term. For languages with a rich morphology such as Bengali, the optimum number of useful feedback terms may be higher compared to English. However, the number of feedback terms can be expected to depend on the type of indexing unit, i.e. sub-words may require additional feedback terms for implicit disambiguation and IR for languages with a rich morphology may also profit from additional feedback terms.

In consequence, the number of feedback terms used for BRF may have to be adjusted and optimized accordingly when the index contains sub-words. Otherwise, the combination of sub-word indexing and BRF might actually degrade performance.

Table VI. Number of types and terms for the FIRE collection based on indexed documents.

Language	Avg_doclen (tokens)	# Types	Index terms
English	235.88	626,085	29.6M
Bengali	264.02	1,238,288	32.5M
Hindi	424.86	459,798	40.3M
Marathi	269.94	1,823,174	26.8M

Two sets of experiments using BRF are conducted by two methods of adaptation: QE_i and QE_{ii}. In QE_i, the number of relevance feedback terms (T) extracted from the top ranked documents (D) is adapted by the increase in vocabulary size (the number of unique tokens) compared to the English document collection, e.g. $T' = T \cdot (\#types_{L1}/\#types_{L2})$ for $L1 = English$ and $L2 \in \{Bengali, Hindi, Marathi\}$ with $D = 10$ and $T = 20$. In QE_{ii}, the number of relevance feedback terms is adapted relative to the change in the number of index terms compared to indexing word forms, e.g. $T' = T \cdot (\#types_{T1}/\#types_{T2})$ for $T1 = word$ and $T2 \in \{CVC, n\text{-}gram\}$.

The former experiments aim at finding out if a larger vocabulary size in a different language would require a higher number of feedback terms. The latter set of experiments concentrate on finding out how the size and number of indexing units requires changes in the number of feedback terms. The BRF experiments in this paper serve only to identify a general trend. Identifying the optimum number of feedback terms for each language and sub-word indexing method would require many more retrieval experiments and may be task-dependent.

In summary, the research question is to confirm that if more terms are indexed (e.g. sub-words instead of stems), more BRF terms have to be used. These experiments were performed only for indexing methods which affect the number of indexing terms, i.e. n-grams and CVC variants.

5. EXPERIMENTS AND RESULTS

For the experiments described in this paper, the Lucene IR toolkit was employed.[8] Lucene does not (yet) provide support for state-of-the-art IR models or for BRF. Support for the Okapi BM25 model [Robertson et al. 1995; Robertson et al. 1998] and for the corresponding BRF approach (see Equation 2 and 3) was implemented for Lucene by one of the authors of this paper.[9] The BM25 score for a document and a query Q is defined as:

$$\sum_{t \in Q} w^{(1)} \frac{(k_1+1)tf}{K+tf} \frac{(k_3+1)qtf}{k_3+qtf} \tag{2}$$

[8] `http://lucene.apache.org/`

[9] The BM25 model has not been specifically designed for IR on sub-words, but was employed for all experiments to keep this experimental parameter fixed.

where Q is the query, containing terms t and $w^{(1)}$ is the RSJ (Robertson / Sparck-Jones) weight of t in Q [Robertson and Sparck-Jones 1976]:

$$w^{(1)} = \frac{(r + 0.5)/(R - r + 0.5)}{(n - r + 0.5)/(N - n - R + r + 0.5)} \tag{3}$$

where

—k_1, k_3, and b are model parameters. The default parameters for the BM25 model used are $b = 0.75$, $k_1 = 1.2$, and $k_3 = 7$.

—N is the number of documents in the collection.

—n is the document frequency for the term.

—R is the number of documents known or presumed to be relevant for a topic. (For BRF, the number of feedback documents is denoted by D.)

—r is the number of relevant documents containing the term.

—*tf* is the frequency of the term t within a document.

—*qtf* is the frequency of the term in the topic.

—$K = k_1((1 - b) + b \cdot doclen/avg\_doclen)$

—*doclen* and *avg_doclen* are the document length and average document length in index terms, respectively.

For BRF, the number of documents D and the number of feedback terms T were by default set to 10 and 20, respectively. These are numbers in the typical range used in information retrieval experiments. The formula to compute the term selection value (TSV) for a term for the IR experiments described in this paper was proposed by Robertson [1990] and is shown in Equation 4.

$$TSV = \frac{r}{R} \cdot w^{(1)} \tag{4}$$

The following subsections describe the results for IR experiments on the FIRE document collection for each language. The best results are set in bold face. The result tables contain the following columns:

—index type: use words (i.e. *word indexing* using unprocessed raw word forms in inflectional forms), PBS (Porter-based stems), CBS (corpus-based stems), n-grams, n-prefixes, CVC (consonant-vowel-consonant sequences), or variants as the index term type.

—QE?: employ query expansion by BRF, yes (Y) or no (N) .

—D: the number of relevance feedback documents.

—T: the number of relevance feedback terms.

—rel_ret: the number of relevant and retrieved documents in the top ranked 1000 documents.

—MAP: mean average precision.

—GMAP: geometric mean average precision.

—P@10: precision at 10 documents.

—chg: absolute change in MAP, compared to the corresponding baseline.

Significance testing on the IR results was performed by ANOVA (analysis of variance) tests per language, followed by Fischer's least significant difference method, LSD) as a post-hoc analysis. Significant changes for ($p < 0.05$) are indicated by * in the result tables. Note that the baseline of indexing raw word forms is chosen because experiments on corpus-based stemming and query expansion are conducted with varying parameters. As a second, more competetive baseline, the corresponding experiments combining raw word indexing with BRF was selected (D=10, T=20) for each language. Significant improvements over this baseline are indicated by $^+$. For a fair comparison, absolute increase in MAP compared to the experiments without BRF is also reported in the tables.

5.1 English Results

Results for the monolingual IR experiments on the English FIRE document collection are shown in Table VII. Both stemming approaches, the Porter stemmer and the corpus-based stemmer, outperform the baseline of indexing words in combination with BRF (+12% and +7% MAP). Query expansion significantly increases IR performance in these cases (+25% and +23% MAP). The best performing method for English is indexing using the Porter stemmer. Indexing n-prefixes ($n \in \{4, 5, 6\}$) significantly outperforms the baseline when combined with BRF and performs similar to applying a stemmer (+24% MAP for the best variant). An additional query expansion step improves the GMAP and the number of relevant retrieved documents for 6-prefixes beyond all other results (+5% more relevant retrieved documents and (+37% GMAP). The corpus-based stemmer has only a slightly worse MAP than the Porter stemmer in combination with BRF (0.5763 vs. 0.5832 MAP). Blind relevance feedback increases MAP in all cases.

Adapting the number of feedback terms increases all performance measures for n-gram-based retrieval, but slightly decreases effectiveness for CVC and variants.

5.2 Bengali Results

Results for the monolingual IR experiments on the Bengali FIRE document collection are shown in Table VIII. Indexing 5-prefixes in combination with query expansion yields the highest number of relevant and retrieved documents for Bengali (1799 out of 1863 relevant documents, +13%), the highest MAP (0.4251, +59%, +15% absolute increase), and the highest GMAP (0.3146, +61%). Precision at 10 documents is highest for 4-prefixes in combination with query expansion (0.5080). The corpus-based stemmer combined with query expansion achieves a significantly higher MAP than the baseline experiment (0.4101 vs. 0.2669 MAP, +54%). A significantly higher MAP is achieved for CVC experiments using query expansion and query expansion with an adapted number of feedback terms.

Adapting the number of feedback terms relative to the increase in vocabulary size consistently increases all performance measures for n-gram retrieval, and increases retrieval effectiveness for most CVC variants.

5.3 Hindi Results

Results for the monolingual IR experiments on the Hindi FIRE document collection are shown in Table IX. The highest MAP was achieved by indexing 5-prefixes (0.2704 MAP, +21%, +5% absolute increase), which also led to the highest num-

Table VII. Results for monolingual English retrieval experiments.

Run	Parameters			Results				
ID	index type	QE?	D T	rel_ret	MAP	GMAP	P@10	chg.[%]
E_B0	word	N	- -	3449	0.4681	0.3713	0.6600	
E_B0QE	word	Y	10 20	3621	0.5602	0.4581	0.7320	+9.2
E_S1	PBS	N	- -	3566	0.5240	0.4487	0.7000	
E_S1QE	PBS	Y	10 20	3630	**0.5832***	0.5039	**0.7540**	+5.9
E_S2	CBS	N	- -	3495	0.4995	0.4038	0.6720	
E_S2QE	CBS	Y	10 20	3631	0.5763*	0.4785	0.7480	+7.7
E_N3	3-gram	N	- -	3319	0.4111	0.2730	0.5720	
E_N3QE	3-gram	Y	10 20	3314	0.4492	0.2537	0.6040	+3.8
E_N3QE$_{ii}$	3-gram	Y	10 80	3343	0.4633	0.2520	0.6340	+5.2
E_N4	4-gram	N	- -	3398	0.4326	0.3336	0.6080	
E_N4QE	4-gram	Y	10 20	3404	0.4742	0.3408	0.6620	+4.2
E_N4QE$_{ii}$	4-gram	Y	10 64	3463	0.4933	0.3497	0.6520	+6.1
E_N5	5-gram	N	- -	3287	0.3908	0.2978	0.5800	
E_N5QE	5-gram	Y	10 20	3373	0.4291	0.3201	0.6420	+3.8
E_N5QE$_{ii}$	5-gram	Y	10 50	3397	0.4573	0.3373	0.6640	+6.7
E_N6	6-gram	N	- -	3274	0.3616	0.2617	0.5580	
E_N6QE	6-gram	Y	10 20	3177	0.4046	0.2288	0.5920	+4.3
E_N6QE$_{ii}$	6-gram	Y	10 40	3294	0.4272	0.2455	0.6280	+6.6
E_P3	3-prefix	N	- -	2457	0.3128	0.1140	0.4139	
E_P3QE	3-prefix	Y	10 20	2455	0.3583	0.1171	0.4667	+4.6
E_P4	4-prefix	N	- -	3421	0.4773	0.3591	0.6400	
E_P4QE	4-prefix	Y	10 20	3540	0.5503*	0.4263	0.7240	+7.3
E_P5	5-prefix	N	- -	3545	0.5156	0.4359	0.6760	
E_P5QE	5-prefix	Y	10 20	3617	0.5668*	0.4771	0.7480	+5.1
E_P6	6-prefix	N	- -	3545	0.5220	0.4505	0.6880	
E_P6QE	6-prefix	Y	10 20	**3634**	0.5822*	**0.5094**	0.7320	+6.0
E_CV	CV	N	- -	2771	0.2893	0.0950	0.4420	
E_CVQE	CV	Y	10 20	2897	0.3445	0.1096	0.4840	+5.5
E_CVQE$_{ii}$	CV	Y	10 52	2909	0.3374	0.1077	0.4780	+4.8
E_VC	VC	N	- -	2720	0.2825	0.0902	0.4060	
E_VCQE	VC	Y	10 20	2875	0.3438	0.1026	0.4900	+6.1
E_VCQE$_{ii}$	VC	Y	10 54	2870	0.3388	0.0912	0.4740	+5.6
E_CVC	CVC	N	- -	3459	0.4653	0.3636	0.6140	
E_CVCQE	CVC	Y	10 20	3511	0.5165	0.3849	0.6540	+5.1
E_CVCQE$_{ii}$	CVC	Y	10 52	3519	0.5153	0.3860	0.6560	+5.0
E_VCV	VCV	N	- -	3191	0.4251	0.2766	0.5900	
E_VCVQE	VCV	Y	10 20	3241	0.4810	0.2937	0.6360	+5.6
E_VCVQE$_{ii}$	VCV	Y	10 54	3254	0.4766	0.2889	0.6460	+5.2

ber of relevant retrieved documents (2480, +5%). Precision at 10 documents was highest for 5-prefixes in combination with BRF. In comparison to the retrieval experiments in other languages, GMAP was very low for all experiments (in all experiments less than 0.04); the highest GMAP value was 0.0357 for indexing overlapping 3-grams. The corpus-based stemmer increases the number of topics with a higher AP than the baseline considerably, but not significantly.

Table VIII. Results for monolingual Bengali retrieval experiments.

Run	Parameters				Results				
ID	index type	QE?	D	T	rel_ret	MAP	GMAP	P@10	chg.[%]
B_B0	word	N	-	-	1592	0.2669	0.1958	0.3700	
B_B0QE	word	Y	10	20	1744	0.3696*	0.2547	0.4460	+10.2
B_B0QE$_i$	word	Y	10	40	1759	0.3796*	0.2757	0.4520	+11.3
B_S2	CBS	N	-	-	1686	0.3034	0.2284	0.4100	
B_S2QE	CBS	Y	10	20	1782	0.4101*	0.2851	0.5060	+10.7
B_N3	3-gram	N	-	-	1735	0.3346	0.2418	0.4340	
B_N3QE	3-gram	Y	10	20	1754	0.3897*	0.2653	0.4580	+5.5
B_N3QE$_{ii}$	3-gram	Y	10	77	1762	0.4211*	0.2794	0.5040	+8.7
B_N4	4-gram	N	-	-	1694	0.3044	0.2214	0.4080	
B_N4QE	4-gram	Y	10	20	1726	0.3582	0.2504	0.4340	+5.4
B_N4QE$_{ii}$	4-gram	Y	10	61	1750	0.3929*	0.2739	0.4760	+8.9
B_N5	5-gram	N	-	-	1610	0.2502	0.1744	0.3160	
B_N5QE	5-gram	Y	10	20	1627	0.2991	0.1938	0.3860	+4.9
B_N5QE$_{ii}$	5-gram	Y	10	48	1701	0.3254	0.2203	0.4060	+7.5
B_N6	6-gram	N	-	-	1538	0.2120	0.1471	0.2880	
B_N6QE	6-gram	Y	10	20	1563	0.2502	0.1488	0.3340	+3.8
B_N6QE$_{ii}$	6-gram	Y	10	38	1640	0.2834	0.1773	0.3700	+7.1
B_P3	3-prefix	N	-	-	1571	0.2541	0.1003	0.3340	
B_P3QE	3-prefix	Y	10	20	1660	0.3429	0.1041	0.4220	+8.9
B_P4	4-prefix	N	-	-	1653	0.3339	0.1646	0.4040	
B_P4QE	4-prefix	Y	10	20	1742	0.4209*	0.2153	**0.5080**	+8.7
B_P5	5-prefix	N	-	-	1702	0.3183	0.2341	0.4240	
B_P5QE	5-prefix	Y	10	20	**1799**	**0.4251***	**0.3146**	0.5060	+10.7
B_P6	6-prefix	N	-	-	1673	0.2998	0.2234	0.4060	
B_P6QE	6-prefix	Y	10	20	1775	0.4106*	0.2953	0.4980	+11.8
B_CV	CV	N	-	-	1371	0.2018	0.0871	0.2980	
B_CVQE	CV	Y	10	20	1539	0.2871	0.1084	0.3700	+8.5
B_CVQE$_{ii}$	CV	Y	10	47	1550	0.2882	0.1083	0.3820	+8.6
B_VC	VC	N	-	-	1456	0.2295	0.0976	0.3140	
B_VCQE	VC	Y	10	20	1584	0.3084	0.1230	0.4020	+7.9
B_VCQE$_{ii}$	VC	Y	10	52	1597	0.3203	0.1243	0.3920	+9.0
B_CVC	CVC	N	-	-	1716	0.3193	0.2508	0.4100	
B_CVCQE	CVC	Y	10	20	1795	0.3941*	0.3048	0.4700	+7.5
B_CVCQE$_{ii}$	CVC	Y	10	47	1795	0.4100*	0.3173	0.5060	+9.1
B_VCV	VCV	N	-	-	1639	0.2834	0.2040	0.3860	
B_VCVQE	VCV	Y	10	20	1727	0.3762*	0.2671	0.4640	+9.3
B_VCVQE$_{ii}$	VCV	Y	10	52	1731	0.3799*	0.2625	0.4760	+9.7

For Hindi, query expansion by BRF always reduced the number of relevant and retrieved documents, and almost always decreased MAP. This effect may be caused by the very low initial AP for some topics, which is reflected in the low GMAP and results in expanding the query from noisy documents.

Also, the average document length (in terms) in the Hindi collection is much higher than for the other collections (cf. Table II). This might mean that query expansion could have less effect for longer documents, because more (potentially

non-relevant) context from the document is considered for the feedback term extraction. The converse argument should also be true: less indexing terms and/or smaller documents provide a closer, more limited context for a better term extraction. Kwok and M. Chan [1998] explored a similar idea: They explore word co-occurrence in small windows of text for expanding short queries.

Adapting the number of terms for BRF typically further decreases performance values compared to the corresponding baseline experiment. Sometimes the MAP and precision at 10 documents are slightly improved.

A careful analysis of the feedback term selection and retrieval process revealed no obvious inconsistencies in the retrieval system used. The Hindi collection has the second highest total number of relevant documents (3,436 relevant documents, see Table III). However, there are several Hindi topics with zero relevant assessed documents, i.e. 12 out of 50 topics have no relevant documents in the Hindi document collection. Six other topics have at least one but less than 10 relevant documents. This reduces the total number of useful topics (e.g. for significance testing) and introduces topics for which a performance increase with BRF is less likely, i.e. when $D = 10$, some non-relevant documents are selected for blind relevance feedback. In comparison, for all English topics there are more than 10 documents were assessed as relevant.

To further investigate the cause of this performance drop, two native Hindi speakers were asked to check retrieval log files generated on the Hindi data containing the all query terms and feedback terms with additional logged information (term frequency, *idf* etc.). Neither identified any regular abnormalities or inconsistencies for the Hindi topics in general or for the badly performing topics.

Additional experiments were conducted to find out if spelling errors or orthographic variants caused worse performance. Sometimes variant words are associated with a high TSV, but do not contribute to finding relevant documents. For example, spelling errors occur rarely but will be assigned a high TSV due to a high *idf* factor. This will result in a top ranking for these terms which means that documents containing these spelling errors – which are not necessarily relevant – will obtain a high rank. To limit the effect of these cases, feedback terms were filtered using different frequency thresholds ($n < 50$, $n < 30$, $n < 10$). For all experiments using term filtering, MAP was slightly lower compared to not using BRF. A similar effect is observed for errors introduced by optical character recognition [Lam-Adesina and Jones 2006].

Experiments with the FIRE data described by participants give no clear indication why query expansion for Hindi should result in lower performance. Most participants did not apply BRF or used it in a different experimental setup. Additional experiments on the FIRE 2010 test set did not show similar results for Hindi, i.e. query expansion for Hindi usually improved MAP for similar experiments [Leveling et al. 2010]. Hence, the high number of FIRE 2008 topics with zero or few relevant documents seems to prohibit obtaining meaningful results for IR experiments with query expansion.

5.4 Marathi Results

Results for the monolingual IR experiments on the Marathi FIRE document collection are shown in Table X. Retrieval on 5-prefixes in combination with BRF

Table IX. Results for monolingual Hindi retrieval experiments.

Run	Parameters				Results				
ID	index type	QE?	D	T	rel_ret	MAP	GMAP	P@10	chg.[%]
H_B0	word	N	-	-	2353	0.2238	0.0231	0.3580	
H_B0QE	word	Y	10	20	1841	0.2371	0.0124	0.3960	+1.3
H_B0QE$_i$	word	Y	10	15	1865	0.2331	0.0139	0.3920	+1.0
H_S2	CBS	N	-	-	2322	0.2389	0.0263	0.3780	
H_S2QE	CBS	Y	10	20	1839	0.2444	0.0136	0.4220	+0.6
H_N3	3-gram	N	-	-	2438	0.2542	**0.0357**	0.3840	
H_N3QE	3-gram	Y	10	20	1933	0.2192	0.0141	0.3800	-3.5
H_N3QE$_{ii}$	3-gram	Y	10	59	1742	0.2132	0.0120	0.3900	-4.1
H_N4	4-gram	N	-	-	2315	0.2290	0.0236	0.3400	
H_N4QE	4-gram	Y	10	20	1823	0.2206	0.0074	0.3400	+0.6
H_N4QE$_{ii}$	4-gram	Y	10	46	1673	0.1965	0.0054	0.3440	-0.1
H_N5	5-gram	N	-	-	2193	0.2036	0.0204	0.3140	
H_N5QE	5-gram	Y	10	20	1873	0.1975	0.0067	0.3300	-0.6
H_N5QE$_{ii}$	5-gram	Y	10	36	1760	0.1993	0.0065	0.3520	-0.4
H_N6	6-gram	N	-	-	2137	0.1765	0.0166	0.2820	
H_N6QE	6-gram	Y	10	20	1486	0.1540	0.0050	0.2760	-0.2
H_N6QE$_{ii}$	6-gram	Y	10	30	1457	0.1566	0.0051	0.2800	-0.2
H_P3	3-prefix	N	-	-	2353	0.2357	0.0249	0.3480	
H_P3QE	3-prefix	Y	10	20	2056	0.2168	0.0112	0.3420	-1.9
H_P4	4-prefix	N	-	-	2407	0.2552	0.0260	0.3600	
H_P4QE	4-prefix	Y	10	20	2118	0.2471	0.0132	0.3760	-0.8
H_P5	5-prefix	N	-	-	**2480**	**0.2704**	0.0288	0.4160	
H_P5QE	5-prefix	Y	10	20	2178	0.2690	0.0221	**0.4300**	-0.1
H_P6	6-prefix	N	-	-	2308	0.2329	0.0224	0.3820	
H_P6QE	6-prefix	Y	10	20	1810	0.2355	0.0144	0.4240	+0.3
H_CV	CV	N	-	-	1808	0.1470	0.0076	0.2980	
H_CVQE	CV	Y	10	20	1526	0.1489	0.0025	0.3000	+0.2
H_CVQE$_{ii}$	CV	Y	10	52	1477	0.1446	0.0019	0.2940	-0.2
H_VC	VC	N	-	-	1884	0.1580	0.0110	0.3080	
H_VCQE	VC	Y	10	20	1524	0.1598	0.0038	0.3020	+0.2
H_VCQE$_{ii}$	VC	Y	10	54	1514	0.1674	0.0030	0.3220	+1.0
H_CVC	CVC	N	-	-	2160	0.2349	0.0261	0.3780	
H_CVCQE	CVC	Y	10	20	1799	0.2170	0.0102	0.3980	-1.8
H_CVCQE$_{ii}$	CVC	Y	10	52	1798	0.2138	0.0098	0.3840	-2.1
H_VCV	VCV	N	-	-	2237	0.1883	0.0189	0.3520	
H_VCVQE	VCV	Y	10	20	1682	0.1897	0.0094	0.3400	+0.1
H_VCVQE$_{ii}$	VCV	Y	10	54	1580	0.1906	0.0064	0.3540	+0.2

returns the best performance for most retrieval measures: the highest number of relevant documents (1061 out of 1095 relevant documents, +20%), MAP 0.4253 MAP (+71%, +18% absolute increase), and 0.4340 P@10. GMAP is highest for 5-prefixes alone (0.2301, +90%).

The corpus-based stemmer performs significantly better than the word indexing baseline (+ 45% MAP). Query expansion increases MAP in general, but can decrease MAP for overlapping 3-grams. In contrast, adapting the number of BRF terms significantly increases the MAP (M_N3QE vs. M_N3QE$_{ii}$). Adapting the

Table X. Results for monolingual Marathi retrieval experiments.

Run	Parameters				Results				
ID	index type	QE?	D	T	rel_ret	MAP	GMAP	P@10	chg.[%]
M_B0	word	N	-	-	886	0.2482	0.1213	0.2740	
M_B0QE	word	Y	10	20	1001	0.2655	0.0968	0.3140	+1.7
M_B0QE$_i$	word	Y	10	58	1004	0.2739	0.0970	0.3080	+2.6
M_S2	CBS	N	-	-	961	0.3594*	0.2072	0.3640	
M_S2QE	CBS	Y	10	20	1040	0.3830*	0.1882	0.3760	+2.4
M_N3	3-gram	N	-	-	1047	0.3690*	0.2068	0.3880	
M_N3QE	3-gram	Y	10	20	1044	0.3586*	0.1385	0.3780	-1.4
M_N3QE$_{ii}$	3-gram	Y	10	79	1055	0.3782*	0.1438	0.4200	+0.9
M_N4	4-gram	N	-	-	1030	0.3675*	0.1986	0.3900	
M_N4QE	4-gram	Y	10	20	1031	0.3891*+	0.1465	0.3900	+2.2
M_N4QE$_{ii}$	4-gram	Y	10	63	1039	0.4019*+	0.1665	0.4160	+3.4
M_N5	5-gram	N	-	-	1022	0.3123	0.1491	0.3120	
M_N5QE	5-gram	Y	10	20	972	0.3214	0.0951	0.3220	+0.9
M_N5QE$_{ii}$	5-gram	Y	10	50	975	0.3244	0.0950	0.3420	+1.2
M_N6	6-gram	N	-	-	966	0.2482	0.1106	0.2680	
M_N6QE	6-gram	Y	10	20	931	0.2406	0.0532	0.2700	-0.7
M_N6QE$_{ii}$	6-gram	Y	10	41	933	0.2514	0.0529	0.2780	+0.3
M_P3	3-prefix	N	-	-	976	0.3097	0.1414	0.3280	
M_P3QE	3-prefix	Y	10	20	1016	0.3118	0.0887	0.3440	+0.2
M_P4	4-prefix	N	-	-	990	0.3956*+	0.2157	0.3620	
M_P4QE	4-prefix	Y	10	20	1047	0.4208*+	0.1882	0.4040	+2.5
M_P5	5-prefix	N	-	-	1016	0.3868*	**0.2301**	0.3780	
M_P5QE	5-prefix	Y	10	20	**1061**	**0.4253***+	0.2219	**0.4340**	+3.9
M_P6	6-prefix	N	-	-	978	0.3333	0.1892	0.3440	
M_P6QE	6-prefix	Y	10	20	1042	0.3686*	0.1722	0.3980	+3.5
M_CV	CV	N	-	-	963	0.3136	0.1453	0.3200	
M_CVQE	CV	Y	10	20	993	0.3195*	0.1165	0.3500	+0.5
M_CVQE$_{ii}$	CV	Y	10	46	993	0.3213*	0.1170	0.3440	+0.8
M_VC	VC	N	-	-	976	0.3140	0.1569	0.3220	
M_VCQE	VC	Y	10	20	1012	0.3237	0.1263	0.3520	+0.9
M_VCQE$_{ii}$	VC	Y	10	54	1011	0.3149	0.1232	0.3280	+0.0
M_CVC	CVC	N	-	-	972	0.3343	0.1708	0.3480	
M_CVCQE	CVC	Y	10	20	976	0.3610*	0.1361	0.3700	+2.7
M_CVCQE$_{ii}$	CVC	Y	10	46	978	0.3737*	0.1429	0.3700	+3.9
M_VCV	VCV	N	-	-	997	0.3499	0.1776	0.3480	
M_VCVQE	VCV	Y	10	20	1033	0.3635*	0.1333	0.3780	+1.4
M_VCVQE$_{ii}$	VCV	Y	10	54	1036	0.3653*	0.1321	0.3800	+1.5

number of feedback terms also tends to increase retrieval measures for n-prefixes and CVC variants.

6. CONCLUSIONS AND OUTLOOK

This paper presented monolingual IR experiments on the FIRE document collections in English, Bengali, Hindi, and Marathi. In particular, language-independent corpus-based stemming, sub-word indexing with n-grams, n-prefixes and CVC, and

BRF on sub-words have been investigated in more than 140 retrieval experiments.

The highest MAP for the official FIRE 2008 experiments reported by the participants is 0.5572 MAP for English [Udupa et al. 2008], 0.4719 MAP for Bengali [Dolamic and Savoy 2008], 0.3487 MAP for Hindi [McNamee 2008], and 0.4575 MAP for Marathi [Dolamic and Savoy 2008]. The results for Bengali and Marathi have been obtained by additionally using the narrative (N) part of the topics to formulate queries. In comparison, the best MAP for the experiments described in this paper are 0.5832 for English, 0.4251 for Bengali, 0.2704 for Hindi, and 0.4253 for Marathi.

The corpus-based stemming approach produced significantly better results than the baseline of indexing words. It provides a knowledge-light baseline for IR in languages which have few resources.

Different methods perform best for different languages. Sub-word indexing produced significantly better results compared to the word indexing baseline and can achieve performance similar to stemming. Word prefixes of length 4 or 5 performed typically best for the Indian languages. However, in combination with query expansion based on BRF, n-prefixes typically perform best for Indian languages and outperform the corpus-based stemming approach. These results confirm results reported by McNamee et al. [2009] for word-internal n-grams and word prefixes for European and Indian languages.

Blind relevance feedback increases MAP in most cases, but query expansion for Hindi mostly reduced IR performance – the number of relevant and retrieved documents, MAP, GMAP and initial precision. However, additional IR experiments on the FIRE 2010 collection showed that query expansion for Hindi typically increases MAP compared to experiments without BRF.

Adapting the number of feedback terms depending on the vocabulary size showed a positive trend that retrieval effectiveness increases when more feedback terms are used for larger vocabularies. This can be observed for experiments across languages (e.g. languages with a richer morphology) and for experiments employing multiple indexing units per word form (e.g. sub-words). The assumption that the number of BRF terms should be increased relative to changes in the indexed vocabulary is confirmed by the trend that MAP is mostly slightly higher for experiments on more and smaller indexing units (e.g. n-grams and CVC). Note that also the reverse assumption may be true: if more terms are indexed, e.g. unstemmed word forms or all words without stopword removal, BRF may be likely to return morphological variants of the query terms or high-frequency words such as semi-stopwords. In order to improve performance, a larger set of relevance feedback terms in comparison with retrieval on indexed stems may be required to include more morphological variants of the same word and additional terms lowering the effects of semi-stopwords in the set of feedback terms. This can be seen as a problem similar to document length normalization: some IR models include a document length normalization factor to compensate for short and long documents. If sub-words are indexed, the document length normalization will also compensate for the generally increased document length. However, a similar factor for adjusting the number of BRF terms, compensating for the changes in the number of index terms for different languages or caused by different indexing units, has not yet been introduced.

Future work will continue investigating indexing techniques and BRF for different languages. The stemmer implementation will be extended to become a more advanced morphological analysis tool, e.g. by supporting prefix removal, internal vowel mutations, and determining morphological signatures or paradigms. For the IR experiments in this paper, a fixed set of 50 suffixes was used for stemming. A more advanced approach will be to determine the best cut-off point for the suffix candidate list dynamically. An improved version of the stemmer might also ignore proper nouns in the document collection.

Further experiments on CVC indexing will be performed. CVC has been used to provide a robust indexing method (e.g. for speech transcripts). In the experiments described in this paper, all resources have been preprocessed and normalized, possibly decreasing the effect of CVC indexing. Additional experiments shall investigate performance differences between glyph-based CVC indexing and CVC indexing on phonetic transcriptions of Indian or Asian text.

Finally, different sub-word indexing techniques can be combined to improve retrieval effectiveness. For example, CVC indexing will be combined with indexing stems to obtain a higher MAP. Furthermore, the BM25 parameters (b, k_1, and k_3) should be optimized for retrieval on sub-words, and improved retrieval models on sub-word indexing should be researched.

The blind relevance feedback experiments are based on expanding queries with a conservative number of terms (opposed to a massive query expansion). A relation between the number of relevance feedback terms and the type of indexing unit or indexed vocabulary size seems to be indicated. However, the relation might not be linear and it might not be the same for each language. Further retrieval experiments will explore the optimum number of feedback terms for different languages and types of indexing units.

ACKNOWLEDGMENTS

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (CNGL) project. The authors are grateful to the reviewers for providing immensely useful feedback and suggestions.

REFERENCES

Billerbeck, B. and Zobel, J. 2004. Questioning query expansion: An examination of behaviour and parameters. In *Proceedings of the Fifteenth Australasian Database Conference (ADC 2004)*, K.-D. Schewe and H. E. Williams, Eds. Vol. 27. Australian Computer Society, Inc., Dunedin, New Zealand, 69–76.

Braschler, M. and Ripplinger, B. 2003. Stemming and decompounding for German text retrieval. In *Proceedings of ECIR-03, 25th European Conference on Information Retrieval, Pisa, Italy, April 14–16, 2003*, F. Sebastiani, Ed. Lecture Notes in Computer Science (LNCS), vol. 2633. Springer, Berlin, 177–192.

Buckley, C., Salton, G., Allan, J., and Singhal, A. 1994. Automatic query expansion using SMART: TREC 3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, D. K. Harman, Ed. National Institute for Standards and Technology (NIST), MD, USA, 69–80.

Chen, A. and Gey, F. C. 2004. Multilingual information retrieval using machine translation, relevance feedback and decompounding. *Information Retrieval 7,* 1–2, 149–182.

Chen, A., He, J., Xu, L., Gey, F. C., and Meggs, J. 1997. Chinese text retrieval without using a dictionary. In *SIGIR '97: Proceedings of the 20th Annual International ACM SIGIR Confer-*

ence on Research and Development in Information Retrieval, July 27-31, 1997, Philadelphia, PA, USA. ACM, New York, NY, USA, 42–49.

Dasgupta, S. and Ng, V. 2006. Unsupervised morphological parsing of Bengali. *Language Resources & Evaluation 40*, 311–330.

Dasgupta, S. and Ng, V. 2007. High-performance, language-independent morphological segmentation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, (NAACL HLT 2007), April 22–27, 2007*, C. L. Sidner, T. Schultz, M. Stone, and C. Zhai, Eds. ACL, Rochester, NY, USA, 155–163.

Dolamic, L. and Savoy, J. 2008. UniNE at FIRE 2008: Hindi, Bengali, and Marathi IR. In *Working Notes of the Forum for Information Retrieval Evaluation 2008, December 12–14, 2008*. Kolkata, India.

Foo, S. and Li, H. 2004. Chinese word segmentation and its effect on information retrieval. *Information Processing and Management: an International Journal 40,* 1, 161–190.

Fox, C. 1992. *Lexical analysis and stoplists*. Prentice-Hall, NJ, USA, 102–130.

Gey, F., Buckland, M., Chen, A., and Larson, R. 2001. Entry vocabulary - a technology to enhance digital search. In *HLT'01: Proceedings of the First International Conference on Human Language Technology Research*. Association for Computational Linguistics, Morristown, NJ, USA, 1–5.

Glavitsch, U. and Schäuble, P. 1992. A system for retrieving speech documents. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Copenhagen, Denmark, June 21–24, 1992*, N. J. Belkin, P. Ingwersen, and A. M. Pejtersen, Eds. ACM, New York, NY, USA, 168–176.

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

Guthrie, D., Allison, B., Liu, W., Guthrie, L., and Wilks, Y. 2006. A closer look at skip-gram modelling. In *Proceedings of the Fifth international Conference on Language Resources and Evaluation (LREC 2006)*. Genoa, Italy, 1222–1225.

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

Hedlund, T. 2002. Compounds in dictionary-based cross-language information retrieval. *Information Research 7,* 2.

Keshava, S. and Pitler, E. 2006. A simpler, intuitive approach to morpheme induction. In *PASCAL Challenge Workshop on Unsupervised Segmentation of Words Into Morphemes - MorphoChallenge 2005, April 12, 2006*. Venice, Italy.

Krovetz, R. 1993. Viewing morphology as an inference process. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, June 27 - July 1, 1993*, R. Korfhage, E. Rasmussen, and P. Willett, Eds. ACM, New York, NY, USA, 191–202.

Kwok, K. L. and M. Chan, M. 1998. Improving two-stage ad-hoc retrieval for short queries. In *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 24-28 1998, Melbourne, Australia*. ACM, New York, NY, USA, 250–256.

Lam-Adesina, A. M. and Jones, G. J. F. 2006. Examining and improving the effectiveness of relevance feedback for retrieval of scanned text documents. *Information Processing and Management 42,* 3, 633–649.

Larkey, L. S., Connell, M. E., and Abduljaleel, N. 2003. Hindi CLIR in thirty days. *ACM Transactions on Asian Language Information Processing 2,* 2, 130–142.

Leveling, J. 2009. A comparison of sub-word indexing methods for information retrieval. In *Proceedings of the LWA 2009 (Lernen-Wissen-Adaption), FG-IR*. Gesellschaft für Informatik, Darmstadt, Germany.

Leveling, J., Ganguly, D., and Jones, G. J. F. 2010. DCU@FIRE2010: Term conflation, blind relevance feedback, and cross-language IR with manual and automatic query translation. In *Working Notes of the Forum for Information Retrieval Evaluation 2010, February 19–21, 2010*. Gandhinagar, India.

Leveling, J. and Hartrumpf, S. 2005. University of Hagen at CLEF 2004: Indexing and translating concepts for the GIRT task. In *Multilingual Information Access for Text, Speech and Images: 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004*, C. Peters, P. Clough, J. Gonzalo, G. J. F. Jones, M. Kluck, and B. Magnini, Eds. Lecture Notes in Computer Science, vol. 3491. Springer, Berlin, 271–282.

Lovins, J. B. 1968. Development of a stemming algorithm. *Mechanical translation and computation 11,* 1-2, 22–31.

Lynam, T. R., Buckley, C., Clarke, C. L. A., and Cormack, G. V. 2004. A multi-system analysis of document and term selection for blind feedback. In *Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, November 8–13, 2004*, D. A. Grossman, L. Gravano, C. Zhai, O. Herzog, and D. A. Evans, Eds. ACM, Washington, DC, USA, 261–269.

Majumder, P., Mitra, M., Parui, S. K., Kole, G., Mitra, P., and Datta, K. 2007. YASS: Yet another suffix stripper. *ACM transactions on information systems (TOIS) 25,* 4, 18:1–20.

McNamee, P. 2001. Knowledge-light Asian language text retrieval at the NTCIR-3 workshop. In *Proceedings of the Third NTCIR Workshop on Research in Information Retrieval, Automatic Text Summarization and Question Answering*, K. Oyama, E. Ishida, and N. Kando, Eds. National Institute of Informatics (NII), Tokyo, Japan.

McNamee, P. 2008. N-gram tokenization for Indian language text retrieval. In *Working Notes of the Forum for Information Retrieval Evaluation 2008, December 12–14, 2008*. Kolkata, India.

McNamee, P. 2008. Textual representations for corpus-based bilingual retrieval. Ph.D. thesis, University of Maryland Baltimore County.

McNamee, P. and Mayfield, J. 2007. N-gram morphemes for retrieval. In *Working Notes of the CLEF 2007 Workshop*. Centromedia, Budapest, Hungary.

McNamee, P., Nicholas, C., and Mayfield, J. 2009. Addressing morphological variation in alphabetic languages. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009*, J. Allan, J. A. Aslam, M. Sanderson, C. Zhai, and J. Zobel, Eds. ACM, New York, NY, USA, 75–82.

Ng, K. 2000. Subword-based approaches for spoken document retrieval. Ph.D. thesis, Massachusetts institute of technology (MIT), Department of electrical engineering and computer science.

Oard, D. W., Levow, G.-A., and Cabezas, C. I. 2001. CLEF experiments at Maryland: statistical stemming and backoff translation. In *Cross-Language Information Retrieval and Evaluation, Workshop of Cross-Language Evaluation Forum, CLEF 2000, Lisbon, Portugal, September 21–22, 2000, Revised Papers*, C. Peters, Ed. Lecture Notes in Computer Science (LNCS), vol. 2069. Springer, Berlin.

Ogawa, Y. and Matsuda, T. 1997. Overlapping statistical word indexing: a new indexing method for Japanese text. *SIGIR Forum 31,* Special issue, 226–234.

Paik, J. H. and Parui, S. K. 2008. A simple stemmer for inflectional languages. In *Working Notes of the Forum for Information Retrieval Evaluation 2008, December 12–14, 2008*. Kolkata, India.

Pal, D., Majumder, P., Mitra, M., Mitra, S., and Sen, A. 2006. Issues in searching for Indian language web content. In *iNEWS'08, October 30, 2008*. ACM, Napa Valley, CA, 93–94.

Pingali, P., Jagarlamudi, J., and Varma, V. 2006. WebKhoj: Indian language IR from multiple character encodings. In *WWW 2006, May 23–26*. ACM, Edinburgh, Scotland, 801–809.

Pirkola, A., Keskustalo, H., Leppanen, E., Känsälä, A.-P., and Järvelin, K. 2002. Targeted s-gram matching: A novel n-gram matching technique for cross- and monolingual word form variants. *Information Research 7,* 2.

Porter, M. F. 1980. An algorithm for suffix stripping. *Program 14,* 3, 130–137.

Robertson, S. E. 1990. On term selection for query expansion. *Journal of Documentation 46,* 4, 359–364.

Robertson, S. E. and Sparck-Jones, K. 1976. Relevance weighting of search terms. *Journal of the American Society for Information Science 27*, 129–146.

Robertson, S. E., Walker, S., and Beaulieu, M. 1998. Okapi at TREC-7: Automatic ad hoc, filtering, VLC and interactive track. In *The Seventh Text REtrieval Conference (TREC-7)*, D. K. Harman, Ed. NIST Special Publication 500-242. National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, 253–264.

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., and Gatford, M. 1995. Okapi at TREC-3. In *Overview of the Third Text Retrieval Conference (TREC-3)*, D. K. Harman, Ed. National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, 109–126.

Rocchio, J. J. 1971. Relevance feedback in information retrieval. In *The SMART retrieval system – Experiments in automatic document processing*, G. Salton, Ed. Prentice Hall, Englewood Cliffs, NJ, USA.

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. 2006. Light stemming approaches for the French, Portuguese, German and Hungarian languages. In *Proceedings of the 2006 ACM Symposium on Applied Computing (SAC), April 23–27, 2006*, H. Haddad, Ed. ACM, Dijon, France, 1031–1035.

Schäuble, P. and Glavitsch, U. 1994. Assessing the retrieval effectiveness of a speech retrieval system by simulating recognition errors. In *HLT '94: Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, Morristown, NJ, USA, 370–372.

Shen, X., Tan, B., and Zhai, C. 2005. Context-sensitive information retrieval using implicit feedback. In *SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005*, R. A. Baeza-Yates, N. Ziviani, G. Marchionini, A. Moffat, and J. Tait, Eds. ACM, New York, NY, USA, 43–50.

Sparck-Jones, K., Walker, S., and Robertson, S. E. 2000. A probabilistic model of information retrieval: development and comparative experiments: Part 1. *Information Processing and Management 36,* 6, 779–808.

Udupa, R., Jagarlamudi, J., and Saravanan, K. 2008. Microsoft research at FIRE2008: Hindi-English cross-language information retrieval. In *Working Notes of the Forum for Information Retrieval Evaluation 2008, December 12–14, 2008*. Kolkata, India.

Xu, J. and Croft, B. 1998. Corpus-based stemming using co-occurence of word variants. *ACM transactions on information systems 16,* 1, 61–81.

Xu, J. and Croft, W. B. 1996. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'96, August 18-22, 1996, Zurich, Switzerland (Special Issue of the SIGIR Forum)*, H.-P. Frei, D. Harman, P. Schäuble, and R. Wilkinson, Eds. ACM, New York, NY, USA, 4–11.

Xu, T. and Oard, D. W. 2008. FIRE-2008 at Maryland: English-Hindi CLIR. In *Working Notes of the Forum for Information Retrieval Evaluation 2008, December 12–14, 2008*. Kolkata, India.