Inf Retrieval (2006) 9:399–420
DOI 10.1007/s10791-006-6390-8

Noun phrases in interactive query expansion and document ranking

Olga Vechtomova

Received: 5 July 2005 / Revised: 7 November 2005 / Accepted: 29 November 2005

Abstract The paper presents several techniques for selecting noun phrases for interactive query expansion following pseudo-relevance feedback and a new phrase-based document ranking method. A combined syntactico-statistical method was used for the selection of phrases for query expansion. Several statistical measures of phrase selection were evaluated. Experiments were also conducted studying the effectiveness of noun phrases in document ranking. One of the major problems in phrase-based document retrieval is weighting of overlapping and non-contiguous word sequences in documents. The paper presents a new method of phrase weighting, which addressed this problem, and its evaluation on the TREC dataset.

Keywords Interactive query expansion · Noun phrases · Document ranking

1. Introduction

Phrases received much attention in information retrieval research throughout its history. This interest can be partially attributed to the fact that phrases typically have a higher information content and specificity than single words, and therefore represent the concepts expressed in text more accurately than single terms. Ideally document and query representations should be mapped directly and unambiguously to the underlying concepts conveyed in text. However, at present, this still remains a difficult goal to reach. Most of the leading statistical IR models, such as probabilistic (Robertson and Spärck Jones, 1976) and vector-space (Salton et al., 1975) rely on the use of single terms and are based on strong term independence assumptions to make them computationally tractable. Experimentally these models have consistently demonstrated high performance results with a variety of large test collections in the evaluation exercises such as TREC (Voorhees and Buckland, 2004). Nevertheless, many attempts have been made to introduce phrases into the retrieval process, but so far with mixed

O. Vechtomova (✉)
Department of Management Sciences, Faculty of Engineering, University of Waterloo, Canada
e-mail: ovechtom@engmail.uwaterloo.ca

and inconclusive results. Our motivation for investigating further the use of phrases in IR was driven by several gaps in the previous research: first, no systematic comparison of various statistical methods of phrase selection for query expansion was conducted; secondly, little was done to explore the usefulness of phrases vs. single terms in the process of interactive query expansion; and finally, weighting of overlapping and non-contiguous phrases is one of the major and yet unsolved problems of phrase-based IR. The work reported in this paper is aimed at filling these gaps through a systematic evaluation of phrase-based methods on the TREC dataset.

Phrases, also referred to as Multiword units (MWUs),[1] comprise a wide variety of lexical associations with various degrees of idiomaticity or compositionality, such as named entities ('Tony Blair', 'United Nations'), nominal compounds ('amusement park', 'free kick') and phrasal verbs ('reach out', 'kick the bucket'). Although MWUs can belong to different lexical categories, our focus is on nominal MWUs, primarily because nouns and noun phrases are considered to be more content-bearing than other syntactic categories. Also, there is some evidence from previous research that noun phrases hold more promise for query expansion in IR (Xu and Croft, 1996).

Query expansion is a widely used technique in IR. In automatic query expansion (AQE) additional terms or phrases are added to the original query by the system, whereas in interactive query expansion (IQE) users select terms or phrases manually. Terms and phrases for query expansion can be extracted using statistical or linguistic methods from a variety of sources, the most common being top-ranked documents in the retrieved set (blind or pseudo-relevance feedback) and documents judged relevant by the user in the retrieved set (relevance feedback). Single-term interactive query expansion techniques were extensively evaluated in the past (Beaulieu and Jones, 1998; Ruthven, 2003). Some researchers investigated the use of phrases in IQE (see Section 2.3), however no systematic comparison of different types of phrases in IQE has been conducted so far. In this work we are interested in studying how different types of phrases can help users to interactively enhance their initial search formulation.

This paper has two foci:

1. To investigate the utility of multiword units (MWUs) and noun phrases in interactive query expansion;
2. To study the effectiveness of noun phrases in the document ranking.

The main goal of the first focus of this study was to investigate the following hypotheses:

Hypothesis 1: Nominal MWUs are better candidates for interactive query expansion than single terms.

Hypothesis 2: Nominal MWUs which exhibit strong degree of stability in the corpus are better candidates for interactive query expansion than noun phrases selected by the statistical characteristics of the individual terms they contain.

We used a combined syntactico-statistical approach for selecting nominal MWUs for interactive query expansion. In the first selection pass, noun phrases were obtained using a part-of-speech (POS) tagger and a noun phrase chunker. In the second pass, statistical measures were applied to select strongly bound MWUs. In particular, we have experimented with two statistical measures to select MWUs from text: the *C*-value (Frantzi and Ananiadou,

[1] We will use these terms interchangeably throughout the paper.

1996) and the Log-Likelihood (Dunning, 1993). Selected MWUs were then suggested to the user for interactive query expansion. Techniques developed for the selection of MWUs are presented in Section 3. Experiments investigating the above hypotheses and evaluation results are described in Sections 5–7.

The goal of the second focus of this work is to study the effectiveness of noun phrases in document ranking. We contribute to the previous findings in the field by further analysing the problems of phrase weighting and suggesting new ways of addressing them. The following hypothesis was investigated:

Hypothesis 3: Ranking documents using noun phrases leads to better performance than ranking documents by single terms.

We have developed a new method of phrase-based document ranking, which specifically addresses the problem of weighting overlapping phrases in documents, which in statistical IR models like probabilistic ones (Spärck Jones et al., 2000) leads to the problem of the over-inflation of the document score. The method is described in detail in Section 4.

2. Previous research

2.1. Statistical vs. syntactical phrases

Hypotheses that phrases are better contents discriminators than single terms have been studied since the beginning of research on automated IR in the 60s. Simple statistical co-occurrence based techniques for identification of phrases have always been rivaled by NLP-based techniques. The main considerations in favour of NLP were: (1) it may have better techniques to uncover meaningful linguistic phrases and (2) it can capture the syntactical relationships between words.

Statistical phrases are typically short-span collocations extracted from text using different statistical measures. Syntactical phrases are identified using a variety of NLP methods ranging from simpler techniques such as part-of-speech tagging, aimed at identifying word-sequences of a certain syntactic pattern like adjective + noun, to more complex methods like extended N-grams and shallow syntactic parsing, attempting to discover uniform semantic units underlying various forms of expression.

At the early stages the motivation for research on automatic phrase identification came from the determination to emulate human indexing. The belief was that multiword descriptors of the kind assigned to documents by human indexers are more useful than single terms. One of the early experiments on phrase indexing was carried out by Bely et al. (1970), who used very elaborate NLP techniques to identify instantiations of thesaurus concepts and their semantic relationships in documents. Despite the fact that no retrieval evaluation was conducted, the research suggested that the structure of the descriptors used for indexing was not flexible enough for effective query-document matching. Another historically important piece of research was undertaken by Salton and Lesk (1968), whose technique consisted in identification of thesaurus terms in text supported by syntactic analysis. The comparison of performance results for syntactical phrases and for statistical phrases, showed that there was no performance improvement in using syntactical phrases over simple statistical phrases. Statistical phrases were defined in their experiments as co-occurrences of constituents of thesaurus descriptors in the same sentence.

One of the most comprehensive early evaluations of phrases in IR was undertaken by Fagan (1987, 1989). The main focus of his experiments was systematic evaluation of statistical phrases under different parameter settings, such as distance between their constituents and their frequency values. The evaluation results showed that performance for statistical phrases was in general better than for single terms. He then compared performance for statistical phrases with performance for syntactical phrases, which he obtained using syntactic parsing, stemming and normalisation to head-modifier pairs. The evaluation showed that linguistically-derived phrases gave results similar to or worse than statistically extracted phrases. When he analysed earlier work taking into account his findings, he concluded that the same pattern, statistical phrases $\geq$ syntactical phrases $\geq$ single terms, was evident in all the experiments. The performance gains from the use of statistical phrases obtained in his experiment were in the range of $17\% - 39\%$. He concluded that syntactical phrases gave poor performance because queries and documents did not share exactly the same phrases. Among the reasons for the systems' inability to match documents and queries by syntactic phrases, Fagan pointed at the low collection frequency of the best phrases and the fact that the documents involved were abstracts. Stzalkowski and Perez-Carballo (1999) pointed to another main reason for this, namely, the limited amount of information about the user's information need conveyed by the queries.

It is worthwhile to note that the above earlier studies of phrases in IR were undertaken on rather small collections. The last decade in IR research saw two major changes: (1) statistical models using single term weighting have been refined to achieve very high and robust performances; (2) the size of test collections has grown dramatically. Some of the phrase indexing and search techniques which used to work well with the old retrieval techniques on small collections, no longer give positive results. Fagan's experiments were later replicated by Hull et al. (1997), leading to only marginal performance gains from using syntactical phrases.

Mitra et al. (1997) conducted a large-scale evaluation of both syntactical and statistical phrases. By statistical phrases they understood contiguous bigrams of non-stopwords which occur in at least 25 documents. Syntactical phrases were defined in their experiments as specific POS-tag sequences (e.g. Noun-Noun, Adjective-Noun). Their studies demonstrate that overall both statistical and syntactical phrases have very little effect on performance. Syntactical phrases showed marginally better performance than statistical phrases when used on their own (i.e. without single terms) in retrieval. An interesting finding, which emerged from this study, is that phrases tend to improve precision at lower document cut-off points of the ranked document sets, and have little or no effect on precision at higher cut-off points. They suggested that phrase search may not be a "precision-enhancing technique", but rather a "recall-enhancing technique."

Evans and Zhai (1996) developed an indexing technique using syntactical phrases. They used a hybrid syntactico-statistical method to identify four types of phrases as indexing units: (1) lexical atoms, e.g. "hot dog"; (2) head-modifier pairs; (3) subcompounds and (4) cross-preposition modification pairs. Their method was evaluated against CLARIT as the baseline, and showed moderate improvements in precision at top ranks. Their method primarily focused on identifying linguistically correct and stable phrases for document indexing, and less so on the problems of weighting phrasal indexing units.

2.2. Phrase weighting

We believe that one of the major and yet unsolved problems of phrase-based techniques is weighting. Phrases like single terms vary in their content-discriminating ability, so it may be possible to treat a phrase in the same way as a single term, and calculate, for example, its inverse document frequency (*idf*) in the same manner. However phrases also have other characteristics, which single terms do not have, and which may need to be reflected in their weighting. One of the most prominent characteristics of phrases is the degree of the stability in the corpus. We distinguish the following types of phrases by their stability in the corpus:

1. Combinations of terms which occur only with each other in many document collections, for example "Burkina Faso."
2. Combinations of terms which frequently occur together as a phrase and whose syntactic structure does not permit any changes (i.e. intervening words, change of word order), for example "amusement park," "stainless steel," "acrylic paint." Typically, one or all terms in such phrases may form lexical-syntactic constructions with other terms as well. If the expression has some degree of idiomaticity (i.e. the phrase as a whole has a different meaning than the combination of individual meanings of its parts), for example "Mad Cow Disease," we may not be able to substitute all or some of the words with related or synonymous words without the radical change of meaning. For example we cannot substitute "mad" with "crazy" in the above example.
3. And finally combinations of terms which have strong degree of flexibility, namely allow intervening words, change of word order, substitution of phrase components with synonyms, hypernyms or hyponyms. For example the exact meaning underlying the phrase "animal protection" can be also represented in text as "protection of animals." The word "animal" can be substituted with hyponyms, such as "reptile" or "mammal."

The above categorisation of phrases has the following implications for IR:

– If the search by one term is highly likely to match the entire phrase (which is typically the case with the phrases of the first category and some phrases of the second category above), then applying phrase search techniques will not be useful.
– If we search by a phrase belonging to the third category, it may be beneficial to relax search constraints to accommodate possible lexical-syntactic variations of the phrase. With this category of phrases, it may even be useful to relax search constraints to allow match on terms separated by longer distances, in order to capture topical relations between terms, rather than only phrasal relations.

Can the phrase-search be successfully integrated into the IR models, which were designed for single-term indexing and searching? Strzalkowski (1995) argued that there are three main problems in applying measures designed for single terms, such as *tf.idf*, for term sets containing both single terms and phrases: (1) single terms normally have high within-document frequency, therefore they tend to get higher weight; (2) whereas phrases, which typically have lower within-document frequency, may receive lower weights; (3) when a document contains both a phrase and a single term which is also part of the phrase, inter-term dependencies emerge. It should be noted that the first two problems apply not only to phrases but to rare single terms as well. Strzalkowski developed techniques for handling the first two problems. Some other IR models also handle the first two problems successfully, for example, a probabilistic model of IR (Spärck Jones et al., 2000) avoids problems like these by progressively reducing the contribution made by the repeating occurrences of a query term to the document score, on the assumption of verbosity.[2]

Another question is whether *idf* can be applied to phrases in the same way as to single terms. Pickens and Croft (2000) conducted an extensive analysis of the use of *idf* for weighting phrases, and concluded that phrases behave simply as medium to rare words, with *idf* having

[2] The term frequency effect can be adjusted in BM25 by means of the tuning constant k_1 (Spärck Jones et al., 2000).

the same discriminatory value for them as for the single terms of this category. Therefore, it can be argued that traditional measures like *idf* can be applied to phrases in the same way as to single terms.

However, the third problem pointed at by Strzalkowski remains unsolved. Robertson et al. (2004) also pointed at the same problem: considering that a query may contain both single terms and phrases, and that some of the single terms may also be part of phrases, then the document matching on the phrase will also match on the single term. As a result both the weight of the phrase occurrence and the weight of the term occurrence will contribute to the document score, artificially inflating it. The solution suggested in Robertson et al. (2004) was to subtract the weight of the single term occurring in the query from the weight of the phrase, containing that term, but the solution did not seem to be effective.

In this paper we examine the phrase-weighting problem further and point at yet another problem that needs to be addressed, namely when the query contains two or more phrases which share one/more terms. In particular this situation can happen following query expansion, where the user or the system selects a number of phrases to be added to the original query. Examples of such phrases are: "stainless steel" and "steel manufacturing." If these phrases match the contiguous string "stainless steel manufacturing" in text, then we face a similar problem of over-inflating the document score as pointed at in Robertson et al. (2004). This problem, however, cannot be solved using their technique. We propose a new method of phrase matching and weighting in the document, which attempts to address this problem. The technique is presented in Section 4.

2.3. Use of phrases in interactive query expansion

Phrases can play a useful role in interactive query expansion by helping users to formulate their information need, in particular when the information need is vague, and users do not know what exactly they are trying to find. Marchionini (1992) and Smeaton and Kelledy (1998) have argued that the process of formulating the query is more cognitively demanding on the part of the user than the process of selecting terms and phrases from the list, as the former involves recall, while the latter—recognition. According to cognitive psychology findings, recall is more demanding than recognition. Therefore in real-world search applications users prefer to formulate terse search statements, which tend to produce poor results, and then browse through the retrieved documents, finding more words and phrases and manually reformulating their queries. Automatically extracting related terms/phrases from the documents retrieved by the original query and showing them to the user facilitates this process as the user does not have to go through large amounts of text.

Smeaton and Kelledy (1998) have experimentally studied the usefulness of statistically-selected phrases in interactive query expansion. In particular they compared the effectiveness of user-selected phrases in search with the user-selected single terms and their combinations. They also looked at the differences between these techniques when used by novice and expert searchers. The best results are obtained when phrases are used in combination with single terms. Also phrase-based query expansion tends to be less effective with the novice searcher than the expert searcher.

Anick and Tipirneni (1999) proposed an interactive query refinement technique using phrases called Paraphrase Search Assistant. Their approach consists in identifying terms in the document set retrieved in response to the user's query, which have a high level of lexical dispersion, i.e. occur in a large number of lexical compounds. For each such term they show in a drop-down textbox noun phrases containing it in the retrieved set. The authors analysed the logs of system use by users in real-life search scenarios, which show a high uptake of the system.

Bruza et al. (2000) conducted a user-based evaluation study of three search methods: unassisted query-based search, directory-based search and phrase-based query reformulation assisted search. Their results demonstrate that an interactive system for phrase-based query reformulation leads to higher number of relevant documents found by users than the unassisted query-based search system.

The contribution of our study to the field of interactive query expansion is that we systematically evaluated the effect of different types of phrases and single terms on retrieval performance in the large-scale TREC experimentation settings.

3. Query expansion methods

In this section we describe techniques developed for interactive query expansion using MWUs following blind feedback. The idea of blind (pseudo-relevance) feedback is to take top-ranked documents, retrieved using the original user's query, and extract query expansion terms/phrases from them. Our approach is to extract query expansion phrases from query-biased summaries of the *n* top-ranked documents. We used a method proposed in Vechtomova et al. (2004) of building query-biased summaries which are composed of *m* sentences selected using two main factors: (1) the *idf* weights of the original query terms present in the sentence, and (2) information value of the sentence, i.e. the combined *tf.idf* value of its words.

In our experiments we used 2-sentence summaries of the 25 top-retrieved documents.[3] We then apply Brill's rule-based tagger (Brill, 1995) and the BaseNP noun phrase (NP) chunker (Ramshaw and Marcus, 1995) to extract noun phrases from the document summaries. Multi-word units are then selected from the list of obtained noun phrases using the C-value and the Log-Likelihood. The two subsections below describe these techniques.

3.1. Selection of query expansion phrases using the C-value

MWUs are characterised foremost by relative stability in the corpus. Some of the noun phrases output by the NP chunker are chance word groupings, and not stable MWUs. We were interested in exploring the value of MWUs compared to all noun-phrases in representing useful query expansion concepts to the user. The method of selecting stable MWUs from noun phrases using C-value is outlined below.

Noun phrases output by the NP chunker are ranked by the average *idf* of their constituent terms. For each phrase we generate the list of all phrases that it subsumes, i.e. contiguous or non-contiguous combinations of words in forward order, including the original complete phrase. For each subphrase, the C-value is calculated. The C-value is a measure of stability of an n-gram in the corpus (Frantzi and Ananiadou, 1996). The C-value formula we used is as follows (Vintar, 2004):

$$C\text{-}value(a) = (length(a) - 1)\left(freq(a) - \frac{t(a)}{c(a)}\right) \quad (1)$$

where, $t(a)$—frequency of the phrase a in longer phrases in the corpus; $c(a)$—number of longer phrases in the corpus including a; *freq*(a)—frequency of the phrase a in the corpus; *length*(a)—number of words in the phrase a.

[3] These parameters showed good performance in the past experiments (Vechtomova et al., 2004).

All subphrases for a given phrase are ranked by the *C*-value. The top-ranked subphrase is then used to replace the original phrase in the list of candidate query expansion terms. The original complete phrase may get a higher *C*-value than any of its subphrases, in which case it is kept without changes.

For example, in our experiment, the bigram "World Cup" received the highest *C*-value out of all its subphrases generated from the phrase "grueling IAU 100-kilometer World Cup" and as a consequence was selected for the phrase list.

Some of the original noun phrases may contain intervening modifiers which are too specific. The reason why we considered non-contiguous word combinations is to eliminate such modifiers and to obtain the most stable and recurrent word combinations. The problem, however, is that some of the resulting phrases are too general (e.g. original phrase: *freak training accident*, selected sub-phrase: *freak accident*), or may have weak or no semantic relatedness to the original phrase (e.g., original phrase: *Moroccan born American runner Khalid Khannouchi*; selected sub-phrase: *born American*). As a result we may have strong topic drift and precision loss at the expense of having linguistically correct MWUs. We did not experiment with using only contiguous word combinations, which might help avoid some of the above problems. However, this remains for future work.

The obtained phrases are then ranked by their C-value, top n of which are shown to the user for interactive query expansion. Table 1 shows the 15 top-ranked phrases selected for the TREC topic "Marathon Training."

3.2. Selection of query expansion phrases using the Log-Likelihood ratio

The Log-Likelihood (Dunning, 1993) has been extensively used for the identification of statistically significant word collocations in text and has shown good results for English.

$$\begin{aligned} Loglike(a,b) = 2 \times \big(&\log \theta_1^{s1}(1-\theta_1)^{n1-s1} + \log \theta_2^{s2}(1-\theta_2)^{n2-s2} \\ &- \log \theta^{s1}(1-\theta)^{n1-s1} - \log \theta^{s2}(1-\theta)^{n2-s2}\big) \end{aligned} \quad (2)$$

Table 1 Top 15 subphrases ranked by C-value and the original phrases from which they were derived (topic "Marathon Training")

Selected sub-phrase	Original phrase
World Cup	Grueling IAU 100-kilometer World Cup
Web site	Marathon's web site
San Diego	San Diego Rock Roll Marathon
York City	York City Marathon
Olympic Games	Athens Olympic Games
Training camp	Training camp
World title	World half marathon title Paula Radcliffe
Athens Olympics	Athens Olympics
Medical Association	International Marathon Medical Directors Association
World Athletics	World Masters Athletics
Training Center	Duoba National Plateau Training Center
Olympic team	Olympic marathon team Athletics Kenya
Training base	Altitude training base
World's fastest	World's fastest
Road Race	25-kilometer 10-kilometer Road Race

Where:

$$s1 = f(a,b) \quad s2 = f(b) - f(a,b)$$
$$n1 = f(a) \quad n2 = N - f(a)$$
$$\theta_1 = \frac{s1}{n1} \quad \theta_2 = \frac{s2}{n2}$$
$$\theta = \frac{f(b)}{N}$$

N—number of words in the corpus; *f*(*a, b*)—frequency of words *a* and *b* appearing together in text; *f*(*a*)—frequency of *a*; *f*(*b*)—frequency of *b*.

The phrase weighting is done as follows: first, from each phrase output by the NP chunker all contiguous bigrams are derived. For each bigram, its Log-Likelihood score is calculated using the Ngram Statistics Package (Banerjee and Pedersen, 2003). The highest Log-Likelihood score of any bigram derived from the phrase is taken as the phrase weight. Top *n* phrases ranked using this weighting scheme are shown to the user for interactive query expansion. This is a rather crude phrase weighting method, but it does reward phrases which contain a strongly bound collocation.

4. Phrase-based document retrieval

Intuitively, searching by phrases, rather than by their constituent terms should lead to better precision. One problem associated with the use of phrases in a statistical IR model, such as probabilistic [2] is that some terms may occur in multiple phrases. For example, assume that there are two phrases in the expanded query: "*air traffic*" and "*traffic control*," and two documents: the first containing one phrase "*air traffic control*," and the second – two phrases "*air traffic*" and "*traffic control*." How should they be weighted? If we calculate weights of each phrase in the document separately and then add them up to get the document score, as is currently done in the probabilistic model for single terms, then both documents would get equal scores. Intuitively, this should not be the case. But then how should the phrase weight be calculated for the first document? The situation gets more complex if we allow for non-contiguous word combinations, i.e. matching the following: "1 *air* 2 *traffic* 10 *control*" (where numbers denote positions of the words in text). Allowing match on non-contiguous word combinations is good for recall as it relaxes search constraints, but the distance between the phrase elements needs to be reflected in the phrase weight. Therefore, the two main issues to be addressed by the phrase search algorithm are:

- remove the problem of overlapping phrases;
- reflect the distance between the phrase elements in the phrase weight.

We have developed a phrase-based document ranking algorithm, which attempts to address the above problems. The algorithm takes a query containing phrases, a set of documents, retrieved by single terms from these phrases, and re-ranks it by using phrase information. We evaluated this algorithm by using queries, consisting of phrases selected by users in the process of interactive query expansion. The remaining part of this section provides a description of the algorithm, and Sections 5 and 7 give the details of its evaluation.

Fig. 1 Subphrases for the phrase "stainless steel manufacturing"

```
stainless steel manufacturing
stainless steel
steel manufacturing
stainless manufacturing
```

Fig. 2 An example of windows extracted from a document

```
# 106 implementation # 120 practical
# 120 practical # 186 implementation
# 4 implementation
# 21 implementation
# 43 implementation
# 59 implementation
```

The first step is to retrieve a set of documents using a best-match document retrieval function[4] and a query consisting of single terms extracted from the query phrases. The next step is to re-rank these documents by using phrases. We take the top 1000 documents per topic in the retrieved set, stem the terms in each document and create a document representation, consisting only of the stemmed occurrences of terms from the query in their original order and their sequential position number in text.

For each query phrase, all possible contiguous and non-contiguous subphrases, including the original phrase, are recorded in a list ranked in descending order of their length. For example, for the query phrase "stainless steel manufacturing," the subphrases are shown in Fig. 1.

For each subphrase in the list we use a pattern matching program *cgrep* (Clarke and Cormack, 1995) to extract the minimal spans of text in the document containing all the words of the subphrase in any order. Such minimal matching strings may only contain one occurrence of each word of the query phrase, for example, *cgrep* would return "`# 106 implementation # 120 practical`," but not "`# 59 implementation # 106 implementation # 120 practical`." Each time cgrep returns a matching string, it is recorded and removed from the document representation, and the procedure is repeated with the same phrase until no matching string is found, in which case the program attempts to match the next phrase in the list, and so on. We refer to the matching strings containing query phrases as *windows*. An example of extracted windows for the phrase "practical implementation" is given in Fig. 2 (the number following the '#' sign is the sequential position of the following word in the original document text).

As it can be seen, windows extracted using the above method might overlap. Our approach to eliminating overlaps in windows is a two-step process: (1) rank the windows by their weight (Section 4.1) and (2) remove overlapping words from the lower ranked windows (Section 4.2).

4.1. Window weighting

For each query phrase, the sets of windows in the document containing exactly the same phrase elements, but possibly within different spans and in different order, are grouped into *bins*. For example, the windows extracted for the query phrase "practical implementation" (Fig. 2) are grouped into two bins, as shown in Fig. 3.

[4] In our experiments, which are described in Sections 5 and 7, we used the Okapi BM25 search function (Spärck Jones et al., 2000).

Fig. 3 An example of windows grouped into bins

```
Bin 1:
 # 106 implementation # 120 practical
 # 120 practical # 186 implementation
Bin 2:
 # 4 implementation
 # 21 implementation
 # 43 implementation
 # 59 implementation
```

All windows in each bin receive the same weight *BinWindowWeight*, which is calculated using one of the following two methods:

Method 1: the actual *idf* of the query phrase contained in the window.

$$BinWindowWeight = idf_{\text{phrase}} \quad (3)$$

The idf_{phrase} score is calculated using the following method: first, Okapi *"sames"* operator is used to find the number of documents containing all words from the query phrase in any order within the same sentence; if no such documents exist, *"and"* operator is used to find the number of documents containing all words from the query phrase in any order and proximity. The "and" operator represents a more relaxed matching criterion, which is needed because there may exist windows with terms, separated by large spans, and never occurring in the same sentence.

Method 2: the sum of *idf* values of all words constituting the query phrase instance in the window.

$$BinWindowWeight = \sum_{i=1}^{n} idf_i \quad (4)$$

Where, n—number of words in the query phrase in the window.

4.2. Removing duplicate windows

After the windows are ranked using one of the window weighting methods described above, we remove overlapping words by doing pairwise comparison of all windows. If two windows have overlapping word(s), these words are removed from the lower ranked window. The windows shown in Fig. 2 after the removal of overlapping words are illustrated in Fig. 4.

```
# 106 implementation # 120 practical
# 4 implementation
# 21 implementation
# 43 implementation
# 59 implementation
# 186 implementation
```

Fig. 4 An example of windows after the removal of overlapping words

All windows extracted from the document for every query phrase are then added to the same list, weighted using Eq. (3) or (4), and the overlapping words are removed as described above. For each window we also keep the index of the phrase which was used to extract it.

4.3. Calculating document scores

The document score is based on the weight of the query phrases it contains. Our approach to query phrase weight calculation is inspired by the term weighting approach in Spärck Jones et al. (2000). The weight of each phrase is calculated using Eq. (5).

$$PhraseWeight = \sum_{n=1}^{|bin|} \left(\frac{(k+1) \times wf_n}{k \times NF + wf_n} \times BinWindowWeight_n \right) \quad (5)$$

where, wf_n is the window frequency in the bin n, which is expressed in Eq. (6) and explained in more detail further in this section; $BinWindowWeight_n$ is the weight of the windows in bin n, calculated using one of the methods described in Section 4.1 above; k is the window frequency normalisation factor, which moderates the contribution of the weight of frequent windows. If $k = 0$ *PhraseWeight* becomes the sum of *BinWindowWeights,* if k is large, then the weight is nearly linear to *wf*; *NF* is the document length normalisation factor that is calculated in the same way as in the BM25 document ranking function (Spärck Jones et al., 2000) and is expressed in Eq. (7).

$$wf = \sum_{w=1}^{|window|} \frac{1}{span_w^p} \quad (6)$$

where, $span = pos(l) - pos(f)$, where: $pos(l)$ – position number of the last query term in the window w and $pos(f)$ – position number of the first query term in the window w; p is a tuning parameter to adjust the effect of span on *wf*. In our experiments $0.1 \leq p \leq 0.5$ gives best results (see Section 7).

If the query phrase matches only one term in the document, the span is set to 1.

$$NF = (1 - b) + b \times \frac{DocLen}{AveDocLen} \quad (7)$$

Where, b is a tuning constant,[5] *DocLen* is the document length in word counts; *AveDocLen* is the average document length in the document collection.

Instead of counting the actual frequency of windows in the bin to get window frequency (*wf*), we adjust the window count by the window span, thus getting a *pseudo-frequency* value. The idea is that the further the words in the window are from each other, the less important this window is considered to be. So, a window containing 2 adjacent words will contribute 1 to the window frequency count, whereas a window containing 2 words within the span of 2, will contribute 0.5 (with $p = 1$). The idea of using pseudo-frequency weights in the Eq. (4) above was inspired by a recent work on weighting terms occurring in documents with multiple fields (Robertson et al., 2004), which proposes a method for weighting term frequencies based on the importance of the document field in which they occur.

[5] It has been experimentally determined in Spärck Jones et al. (2000) that $b = 0.75$ gives best results on TREC data.

Document matching score (MS) is calculated as the sum of PhraseWeights of all query phrases in the document, and is expressed in Eq. (8), where $|phrase|$ is the number of phrases occurring in the document.

$$MS = \sum_{i=1}^{|phrase|} PhraseWeight_i \quad (8)$$

The MS is applied to re-ranking of the top 1000 documents retrieved using single terms from the query phrases.

The proposed method was evaluated on the dataset of the High Accuracy Retrieval from Documents (HARD) track of TREC 2004. The evaluation results are presented in the next section.

5. Evaluation

The testbed for our experiments is the Okapi IR system, which is based on the Robertson/Spärck Jones probabilistic model of retrieval (Spärck Jones et al., 2000). The evaluations of the developed techniques were conducted within the framework of the HARD (High Accuracy Retrieval from Documents) track of TREC 2004 (Allan, 2005). The HARD track evaluation framework includes an interactive component, which allowed us to test interactive query expansion techniques. The interactive evaluation experiment consists of the following steps:

1. TREC organisers release the search statements (topics) formulated by the annotators (users) in the traditional TREC format (Title, Description and Narrative) to the participating sites.
2. Participating sites use any information from the topics to produce the initial (baseline) document sets and compose clarification forms for the user to fill in. The purpose of clarification forms is to clarify or refine the annotator's search statement.
3. The annotator (user) fills out clarification forms (with a 3-minute time limit per form).
4. Participating sites use the annotator's feedback to the clarification forms to improve the search (for example by query expansion). The end result is a new document set.
5. The annotator performs relevance judgements of the retrieved sets.[6]

The HARD track test collection includes the document corpus (635,650 documents from eight newswire collections) and 50 topics. In addition to the traditional TREC topic fields of Title, Description and Narrative, the topics also contained several Metadata fields, describing various additional search criteria, such as "genre," "retrieval element" and "familiarity." We did not use any of the metadata in the runs reported here except "retrieval element," which takes two values "Document" or "Passage." In all expansion runs for topics with the retrieval element "Document" we used the Okapi document retrieval function BM25, and for topics with the retrieval element "Passage" we used the Okapi passage retrieval function BM250.

We conducted two baseline runs using only the information available in the TREC topics: in the run *baseTD*, we used all non-stopword terms extracted from the Title and Description

[6] Top 75 documents from two runs per site were added to the relevance judgement pool. Each document in the pool was assigned a binary relevance judgement. The same annotator who formulated the topic provided feedback to all clarification forms for that topic and performed relevance judgements.

fields of the topic and in *baseT*, we used all terms from the Title field only. For both runs we applied Okapi BM25 search function.

Four clarification forms were generated for each topic. Phrases for each clarification form were extracted from 2-sentence query-biased summaries (Vechtomova et al., 2004) of the top 25 documents retrieved in the run *baseTD*, as Title+Description gave higher performance than Title on HARD 2003 data.

- *1st clarification form:* top n phrases selected using the C-value method (Section 3.1 above);
- *2nd clarification form:* single terms from the phrases displayed in the 1st clarification form;
- *3rd clarification form:* top n phrases output by the NP chunker and ranked by the average *idf* of their constituent terms;
- *4th clarification form:* top n phrases selected using the Log-Likelihood ratio (Section 3.2 above).

We used the 2nd clarification form to investigate whether users select better terms when they are shown in the context of phrases (in the 1st clarification form), than separately. By comparing the phrases selected from the 3rd clarification form with the 1st and 4th we aim to answer the question whether the application of the measures of phrase stability in the corpus leads to better phrases for query expansion.

Four query expansion runs were conducted. Runs ExpSingle$\{1, 2, 3, 4\}$ used the feedback provided by the users to the 1st, 2nd, 3rd and 4th sets of clarification forms respectively. In each run the query was constructed by splitting the phrases selected by the user from the corresponding clarification form into single terms and adding them to the original query terms. Each term in the expanded query was weighted in Okapi using pseudo-relevance data.[7] The BM25/BM250 search function was then used to search the database. The results of the evaluation of the single-term search using feedback from the four clarification forms are presented in the next section. Following the TREC experiments we have developed the phrase-search method, presented earlier in this paper, which is an improvement of a technique used in TREC (Vechtomova and Karamuftuoglu, 2005). Here for each topic we take the top 1000 documents retrieved in the run ExpSingle1 (i.e. using single terms from the user-selected phrases from the 1st clarification form) and re-rank them using the method presented in Section 4. The evaluation results of the phrase-based search algorithm are discussed in Section 7.

6. Evaluation of the single-term search using clarification form feedback

The results of the evaluation of the query expansion methods using clarification forms are presented in Table 2. All expanded runs significantly improve the performance over the baseline run BaseTD (*t*-test at .05 significance level).

Average Precision of the expanded queries containing phrases selected from clarification form 1 is 5% higher than that of the queries containing single terms selected from clarification form 2. The difference is not statistically significant (*t*-test at .05 significance level), however it suggests that showing terms in the context of phrases somewhat helps users to select better terms, compared to showing single terms to them. This provides support for Hypothesis 1. On average users selected 21 phrases from the 1st clarification form and 27 single terms from the 2nd form. There were 675 phrase-terms selected only from the 1st form, 384 terms selected only from the 2nd form and 921 terms selected from both forms.

[7] The top 25 documents retrieved in the *baseTD* run were assumed to be relevant.

Table 2 Results of the runs, averaged over 45 topics[8]

Run	P@5	P@10	AveP	R-Precision
Title terms (BaseT)	0.3556	0.3089	0.2196	0.2499
Baseline, Title + Description (BaseTD)	0.48	0.42	0.2693	0.3011
Single-term search, Query expansion with phrases from clarification form 1 (ExpSingle1)	0.5022 (+4.6%)	0.4889 (+16%)	0.3176 (+18%)	0.3381 (+12.3%)
Single-term search, Query expansion with terms from clarification form 2 (ExpSingle2)	0.5111 (+6.4%)	0.48 (+14.3%)	0.3026 (+12.4%)	0.3283 (+9%)
Single-term search, Query expansion with phrases from clarification form 3 (ExpSingle3)	0.5111 (+6.4%)	0.4911 (+16.9%)	0.3191 (+18.5%)	0.3352 (+11.3%)
Single-term search, Query expansion with phrases from clarification form 4 (ExpSingle4)	0.4978 (+3.7%)	0.4689 (+11.6%)	0.3019 (+12.1%)	0.3256 (+8.1%)

There is negligible difference between the performance of the queries formed from the phrases selected using the average *idf* of their terms (ExpSingle3) and queries from the phrases selected using the measures of phrase stability in the corpus: the C-value (ExpSingle1) and the Log-Likelihood ratio (ExpSingle4). This suggests that the statistical component of phrase selection does not play an important role when it is combined with syntactical phrase selection techniques, such as POS-tagging and NP-chunking. Hypothesis 2 is, therefore, not supported. In the next section we discuss the evaluation of the phrase search algorithm.

7. Evaluation of the phrase search algorithm

The set of documents retrieved by single phrase-terms extracted from the user-selected phrases (ExpSingle1) was re-ranked using the proposed phrase-search method. The run ExpPhrase1 uses the *idf* of the phrase in the window to calculate *BinWindowWeight* (Eq. (3)), and ExpPhrase2 uses the sum of *idf* of phrase-terms (Eq. (4)). The average number of words in the user-selected phrases is 2, and out of all query phrase occurrences in the documents, 46% occur in adjacent positions (span 1), 59.52% occur within the span of 5 words, and 67.51% – within the span of 10 words. We evaluated our algorithm without imposing a span limit (no-span-limit runs) and with setting a span limit on matching phrases to 1 (adjacency), 5 and 10 words. Table 3 shows the results for the best runs (the k and p values shown are those that yielded the best Mean Average Precision).

The best results were obtained by using the sum of *idf* of phrase-terms in calculating *BinWindowWeight*, with no span limit (ExpPhrase2_no-span-limit run with $p = 0.1$ and $k = 0.75$). Average Precision improved by 5.6%, and Precision at 5 documents improved by

[8] Five out of 50 topics had no relevant documents, and were excluded from the official HARD evaluation.

Table 3 Results averaged over 45 topics of HARD track 2004

Run	P@5	P@10	AveP	R-Prec
Single-term search, (ExpSingle1), baseline	0.5022	0.4889	0.3176	0.3381
ExpPhrase1_no-span-limit ($p = 0.2; k = 0.75$)	0.5111 (+1.8%)	0.4756 (−2.7%)	0.3276 (+3.2%)	0.34 (+0.5%)
ExpPhrase1_adj ($k = 1$)	0.5244 (+4.4%)	0.4689 (−4%)	0.3205 (+0.9%)	0.3302 (−2.3%)
ExpPhrase1_span-5 ($p = 0.5; k = 0.75$)	0.5067 (+0.9%)	0.4556 (−6.8%)	0.3201 (+0.8%)	0.3204 (−5.2%)
ExpPhrase1_span-10 ($p = 0.1; k = 0.75$)	0.5156 (+2.7%)	0.4778 (−2.3%)	0.3141 (−1.1%)	0.3254 (−3.8%)
ExpPhrase2_no-span-limit ($p = 0.1; k = 0.75$)	0.5156 (+2.7%)	0.4733 (−3.2%)	0.3354 (+5.6%)	0.3346 (−1%)
ExpPhrase2_adj (k=0.75)	0.5156 (+2.7%)	0.4622 (−5.7%)	0.3262 (+2.7%)	0.3258 (−4%)
ExpPhrase2_span-5 ($p = 0.1; k = 0.75$)	0.5067 (+0.9%)	0.4556 (−7.3%)	0.3255 (+2.5%)	0.323 (−4.5%)
ExpPhrase2_span-10 ($p = 0.1; k = 0.75$)	0.5244 (+4.4%)	0.4756 (−2.8%)	0.3295 (+3.7%)	0.3207 (−5.4%)

2.7%, however Precision at 10 documents dropped by 3.2%. The results of runs setting a span limit on matching phrases were overall not as good as without a span limit on all measures, except Precision at 5 in the ExpPhrase2_span-10 run. The fact that the best results were obtained with no span limit suggests that our phrase matching model offers some benefits over phrase-matching models based on strict adjacency. Limiting the span to 5 or 10 words also leads to smaller improvements. It is noteworthy to mention that the best results were obtained with $0.1 \leq p \leq 0.5$, which suggests that some reduction of the phrase weight with the increase of the span between its elements offers additional relevance-discriminating power.

Figure 5 shows the comparison of the best phrase run ExpPhrase2_no-span-limit with the single-term run ExpSingle1. Out of 45 topics, 26 had higher AveP with the phrase-search algorithm, and 19 – lower; 13 had higher P@10, and 12 lower; and 12 had higher P@5, whereas 8–lower. As can be seen from Fig. 3, topic 444 ("European Elections") has unusually lower performance with phrases than with single terms. If we exclude this topic from comparison, then the mean P@10 remains almost unchanged when we use phrases (+0.9%), and the mean P@5 improvement reaches 6.5%.

A more detailed look at some topics, the performances of which were improved or deteriorated, revealed that the use of phrases altered rankings of some documents substantially, for example two relevant documents in topic 420 ("Internet security from quantum computing") were promoted from ranks 48 and 50 to 1 and 2; however in topic 444 ("European Elections") four relevant documents were demoted from top 10 to rank 200 and lower.

Figure 6 shows recall-precision graph of the best baseline, single-term and phrase-based runs. It is evident that while single-term query expansion run mainly improves precision at higher recall levels, the phrase-based run improves precision at lower recall levels. Previous work, e.g. (Mitra et al., 1997), suggested that phrases predominantly lead to improvements in precision at higher-recall levels, which is not supported by our results. Although the improvement obtained here is a moderate one, it indicates that the use of phrases in search may potentially lead to greater improvements in high precision tasks.

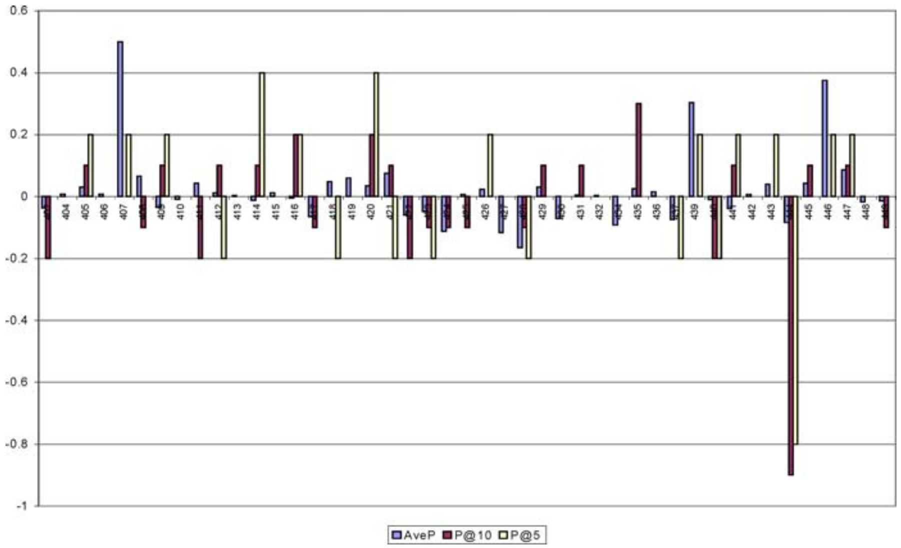

Fig. 5 Comparison of the best phrase-based run ExpPhrase2_no-span-limit with the single-term run (ExpSingle1)

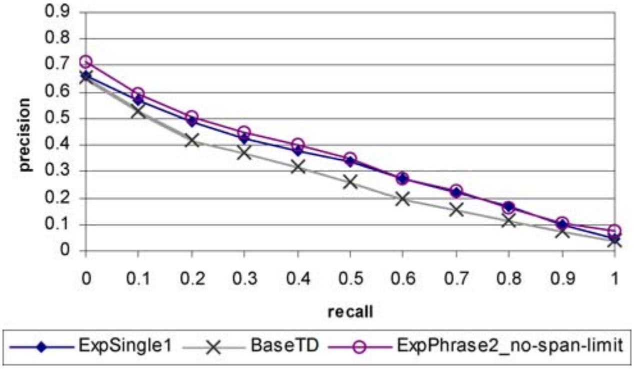

Fig. 6 Recall-precision graph of the best baseline, single-term and phrase-based query expansion runs

8. The effect of user familiarity with the topic on phrase selection and retrieval performance

The familiarity metadata was used in the HARD track to indicate the extent to which the searchers formulating the topic were familiar with it. Out of 45 topics used in the evaluation there were 25 topics with user familiarity "little" and 20 topics with familiarity "much."

We have analysed the effect of the searcher familiarity with the topic on two variables:

- the number of phrases selected for query expansion;
- the performance of different search methods.

We hypothesise that the more familiar the searchers are with the subject of the query, the more phrases they are able to choose for query expansion. Our experimental results

Table 4 The average number of QE phrases/terms selected by users with "little" and "much" familiarity

	Average number of selected phrases/terms		
Clarification form (CF)	Familiarity "little"	Familiarity "much"	Difference (%)
CF1: *C*-value selected phrases	19.6	24.9	27
CF2: Single terms from CF1	24	36	49
CF3: Ave. IDF selected phrases	15	25	67.5
CF4: Log-likelihood selected phrases	19.7	29.6	50.3

Table 5 Mean average precision of topics formulated by users with "little" and "much" familiarity

	AveP		
Run	Familiarity "little"	Familiarity "much"	Difference (%)
Title terms (BaseT)	0.184	0.265	44.2
Baseline, Title + Description (BaseTD)	0.228	0.320	40.7
Single-term search, Query expansion with phrases from clarification form 1 (ExpSingle1)	0.266	0.382	43.8
Single-term search, Query expansion with terms from clarification form 2 (ExpSingle2)	0.269	0.345	28.4
Single-term search, Query expansion with phrases from clarification form 3 (ExpSingle3)	0.280	0.368	31.2
Single-term search, Query expansion with phrases from clarification form 4 (ExpSingle4)	0.251	0.366	45.8
ExpPhrase2_no-span-limit ($p = 0.1$; $k = 0.75$)	0.286	0.397	38.5

support this hypothesis. In all four clarification forms users familiar with the topic selected substantially more QE terms and phrases than the less familiar users (Table 4). The difference observed in all clarification forms but one, CF1 (*C*-value selected phrases), was statistically significant (using *t*-test at .05 significance level).

Next, we hypothesise that the more familiar the searchers are with the topic, the better the performance of their unexpanded and expanded queries should be. The results of all baseline and experimental runs support this hypothesis: in all runs topics with "much" familiarity show higher Mean Average Precision, as evident from Table 5. The table also includes the best phrase-based document re-ranking run (ExpPhrase2_no-span-limit).

The analysis of search results by familiarity reveals interesting patterns in the performance of the phrase-based document re-ranking method. By analysing topics with different familiarity levels, we can see that phrase-based document re-ranking (run ExpPhrase2_no-span-limit in Table 5) improves the Average Precision of topics with "little" familiarity by 7.5%, and the AveP of topics with "much" familiarity by 3.9% compared to the single-term document ranking ExpSingle1 (Table 5). It is hard to say what exactly contributes to a better performance of the phrase-search algorithm with "little" familiarity topics, but it could be simply because there is more scope for improvement compared to "much" familiarity topics.

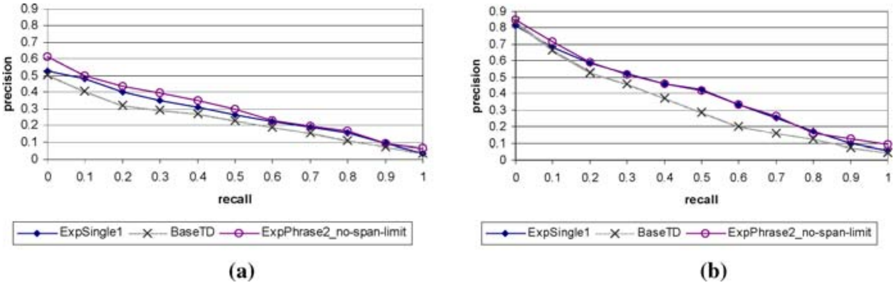

Fig. 7 Recall-precision graphs of the runs for topics with (a) familiarity "little" and (b) familiarity "much"

As can be seen from Fig. 7 , topics with "little" familiarity, and to a lesser extent topics with "much" familiarity, have somewhat higher precision at lower recall levels than the baseline ExpSingle1. This suggests that phrase-based search techniques might be appropriate for high-precision tasks.

9. Conclusions and future work

In this paper we presented a comparative evaluation of different phrase selection techniques in interactive query expansion and a novel phrase search method. A combined syntactico-statistical method was used for the selection of phrases. First, noun phrases were selected using a part-of-speech tagger and a noun-phrase chunker, and secondly, different statistical measures were applied to select phrases for query expansion.

The following three hypotheses were investigated in this study:

Hypothesis 1: Nominal MWUs are better candidates for interactive query expansion than single terms.

We studied whether users select better terms when they are shown to them in the context of phrases, than separately. The users were asked to select query expansion items from two clarification forms: one with the complete phrases selected by the C-value, and the other with the single terms from these phrases. The results suggest that showing phrases to the users helps them to select somewhat better query expansion terms, than showing single terms.

Hypothesis 2: Nominal MWUs which exhibit strong degree of stability in the corpus are better candidates for interactive query expansion than noun phrases selected by the statistical characteristics of the individual terms they contain.

We evaluated three statistical phrase selection methods: the C-value, Log-Likelihood ratio and average *idf* of phrase terms. Phrases selected using these methods were shown to the user in clarification forms for interactive query expansion. Evaluation experiments did not demonstrate substantial difference between these statistical methods in their effect on retrieval performance.

Hypothesis 3: Ranking documents using noun phrases leads to better performance than ranking documents by single terms.

We have developed a method for phrase-based document ranking, which addresses the problems of weighting of overlapping and non-contiguous word sequences in documents. Some improvements in average precision and precision at 5 documents were obtained through the use of phrases over the use of single terms. Weighting phrases by using the sum of *idf* of phrase components and by using the actual *idf* of the phrase as a string of words occurring in the same sentence lead to comparable results, although the sum of *idf* of phrase components was slightly better. The best values of p (the constant adjusting the effect of span) range between 0.1 and 0.5, which suggests that the phrase weight should not decrease linearly with the span, and also that the use of span in phrase weighting leads to some, albeit moderate, improvements. Our results demonstrate that although, overall improvements from using phrases are moderate, which is consistent with past work, some improvements in average precision, and at top ranks, namely precision at 5, can be obtained. This is particularly important for high accuracy retrieval tasks. Previous work has demonstrated comparable improvements in average precision, but gains were predominantly due to improved precision at lower ranks (Mitra et al., 1997).

We also investigated the performance of the phrase search method on topics, formulated by users with "little" and "much" familiarity. The results indicate that topics with "little" familiarity benefit more from the phrase search, especially at low-recall levels. It was also found that users with "much" familiarity select substantially more query expansion phrases from clarification forms than users with "little" familiarity.

One of the issues with any phrase-search algorithms is the increased computational time needed to calculate phrase weights, compared to single-term weighting schemes, so any benefit obtained through the use of phrases should be weighed against the extra computational cost required. We have adopted a late-binding strategy, whereby terms are combined to form phrases at search time when the query is known, as opposed to pre-computing the phrases at indexing time. This has an advantage of tuning the document representation to the particular query, albeit at the cost of possibly longer query processing time.

One of the possible future extensions of this work will be to use measures of phrase stability to estimate phrase weight in the documents. Phrases differ by their stability in the corpus, as discussed earlier in the paper, therefore it may be advantageous to weight them on the basis of their stability. For example, a document which has a partial match on a non-compositional or idiomatic phrase (e.g. "Salt Lake City") is more likely to be non-relevant, than a document that has a partial match on a non-idiomatic expression (e.g. "organic product"). Therefore the weight of the partially matching phrase should be reduced more in the first case than in the second.

References

Allan, J. (2005). HARD Track overview in TREC 2004. High Accuracy Retrieval from Documents. In: E. Voorhees & L. Buckland (Eds.), *Proceedings of the Thirteenth Text Retrieval Conference*, NIST, Gaithersburg, MD, November 2004.

Anick, P. G., & Tipirneni, S. (1999). The Paraphrase search assistant: Terminological feedback for interactive information seeking. In: *Proceedings of the 22nd Annual International ACM SIGIR Conference* (pp. 153–159). Berkeley, California.

Banerjee, S., & Pedersen, T. (2003). The design, implementation and use of the ngram statistics package. In: *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*. Mexico City.

Beaulieu, M., & Jones, S. (1998). Interactive searching and interface issues in the Okapi best match probabilistic retrieval system. *Interacting with Computers*, *10*(3), 237–248.

Bely, N., Borillo, A., Virbel, J., & Siot-Decauville, N. (1970). *Procédures d'analyse sémantique appliquée à la documentation scientifique*. Paris: Gauthier.

Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics*, *21*(4), 543–565.

Bruza, P. D., McArthur, R., & Dennis, S. (2000). Interactive internet search: Keyword, directory and query reformulation mechanisms compared. In: *Proceedings of the 23rd Annual International ACM SIGIR Conference* (pp. 280–287). Athens, Greece.

Clarke, C. L. A., & Cormack, G. V. (1995). On the use of regular expressions for searching text. University of Waterloo Computer Science Department Technical Report number CS-95-07, University of Waterloo, Canada.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, *19*(1), 61–74.

Evans, D. A., & Zhai, C. (1996). Noun-phrase analysis in unrestricted text for information retrieval. In: *Proceedings of the ACL-96, 34th Annual Meeting of the Association for Computational Linguistics*.

Fagan, J. L. (1987). Automatic phrase indexing for document retrieval: An examination of syntactic and non-syntactic methods. In: *Proceedings of the Tenth ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 91–101). New Orleans.

Fagan, J. L. (1989). The effectiveness of a nonsyntatic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, *40*(2), 115–132.

Frantzi, K. T., & Ananiadou, S. (1996). Extracting nested collocations. In: *Proceedings of the 16th Conference on Computational Linguistics* (pp. 41–46). COLING.

Hull, D., Grefenstette, G., Schulze, M., Gaussier, E., Schütze, H., & Pedersen, J. (1997). Xerox TREC-5 site report: Routing, filtering, NLP and Spanish tracks. In: D., Harman (Ed.), *Proceedings of the fifth text retrieval conference* (pp. 167–180). Gaithersburg, MD, November, 1996.

Marchionini, G. (1992). Interfaces for end-user information seeking. *Journal of the American Society for Information Science*, *43*(2), 156–163.

Mitra, M., Buckley, C., Singhal, A., & Cardie, C. (1997). An analysis of statistical and syntactic phrases. In: *Proceedings of RIAO97, Computer-Assisted Information Searching on the Internet*. Montreal, Canada, pp. 200–214.

Pickens, J., & Croft, B. (2000). An exploratory analysis of phrases in text retrieval. In: *Proceedings of RIAO* 2000.

Ramshaw, L., & Marcus, M. (1995). Text chunking using transformation-based learning. In: *Proceedings of the Third ACL Workshop on Very Large Corpora*, MIT.

Robertson, S. E., & Spärck, J. K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, *27*, 129–146.

Robertson, S.E., Zaragoza, H., & Taylor, M. (2004). Simple BM25 extension to multiple weighted fields. In: *Proceedings of the Thirteenth Conference on Information and Knowledge Management*, Washington.

Robertson, S. E., Zaragoza, H., & Taylor, M. (2004). Microsoft Cambridge at TREC-12: HARD track. In: *Proceedings of the Twelfth Text Retrieval Conference,* E., Voorhees, & L., Buckland (Eds.), *NIST* (pp. 418–425), Gaithersburg, MD.

Ruthven, I. (2003). Re-examining the potential effectiveness of interactive query expansion. In: *Proceedings of the 26th ACM-SIGIR conference*. Toronto, Canada, pp. 213–220.

Salton, G., & Lesk, M. E. (1968). Computer evaluation of indexing and text processing. *Journal of the ACM*, *15*(1), 8–36.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for information retrieval. *Communications of the ACM*, *18*(11), 613–620.

Smeaton, A. F., & Kelledy, F. (1998). User-chosen phrases in interactive query formulation for information retrieval. In: *Proceedings of the 20th BCS-IRSG Colloquium*, Grenoble, France, Springer-Verlag Workshops in Computing.

Spärck Jones, K., Walker, S., & Robertson, S. E. (2000). A probabilistic model of information retrieval: Development and comparative experiments. *Information Processing and Management*, *36*(6), 779–840.

Strzalkowski, T. (1995). Natural language information retrieval. *Information Processing and Management*, *31*(3), 397–417.

Strzalkowski, T. & Perez-Carballo, J. (1999). Evaluating natural language processing techniques in information retrieval. In: T. Strzalkowski (Ed.), *Natural language information retrieval* (pp. 113–145). Kluwer Academic Publishers.

Vechtomova, O., & Karamuftuoglu, M. (2005). Approaches to high accuracy document retrieval in HARD track. In: E., Voorhees, & L., Buckland (Eds.), *Proceedings of the Thirteenth Text Retrieval Conference*, NIST, Gaithersburg, MD, November 2004.

Vechtomova, O., Karamuftuoglu, M., & Lam, E. (2004). Interactive search refinement techniques for HARD tasks. In: E., Voorhees, & L., Buckland (Eds.), *Proceedings of the twelfth text retrieval conference* (pp. 820–827). NIST, Gaithersburg, MD, November 2003.

Vintar, Š. (2004). Comparative evaluation of C-value in the treatment of nested terms. In: *Proceedings of MEMURA 2004 Workshop (Methodologies and Evaluation of Multiword Units in Real-world Applications), Language Resources and Evaluation Conference (LREC)* (pp. 54–57). Lisbon, Portugal.

Voorhees, E., & Buckland, L. (2003). In: *Proceedings of the Twelfth Text Retrieval Conference*, NIST, Gaithersburg, MD.

Xu, J., & Croft, B. (1996). Query expansion using local and global document analysis. In: *Proceedings of the 19th International Conference on Research and Development in Information Retrieval (SIGIR '96)* (pp. 4–11). Zurich, Switzerland.