A study of statistical query expansion strategies for sentence retrieval

David E. Losada
Grupo de Sistemas Inteligentes
Departamento de Electrónica y Computación
Universidad de Santiago de Compostela
Campus sur s/n, Santiago de Compostela, Spain
dlosada@dec.usc.es

ABSTRACT

The retrieval of sentences that are relevant to a given information need is a challenging passage retrieval task. In this context, the well-known vocabulary mismatch problem, present in most Information Retrieval processes, arises severely because of the fine granularity of the task. Short queries, which are usually the rule rather than the exception, come to aggravate the problem. Consequently, effective sentence retrieval methods tend to apply some form of query expansion, usually based on pseudo-relevance feedback. Nevertheless, there are no extensive studies comparing different expansion strategies for sentence retrieval problems. In this work we aim to fill this gap. We start from a set of retrieved documents in which relevant sentences have to be found. In our experiments we test different term selection strategies and we also check whether expansion before sentence retrieval can yield reasonable performance. This is particularly novel because expansion techniques for sentence retrieval are often applied after a first retrieval of sentences and there are no comparative results available between expansion before and after sentence retrieval. This comparison is valuable not only for testing distinct expansion-based methods but also because there are important implications in time efficiency.

Keywords

Sentence retrieval, Query expansion, Information Retrieval

1. INTRODUCTION

The availability of effective sentence retrieval methods is potentially beneficial to many IR systems. There are many tasks whose performance is affected by the effectiveness of a sentence retrieval module. In web IR, information access can be facilitated provided that a good ranking of sentences, ordered by estimated relevance to the user, is supplied [18]. Question answering systems usually require some form of passage retrieval to isolate the document pieces in which the answer is likely to be found. This step is often done at the sentence level [11]. One of the main areas in text summarization is centered on building summaries by extracting important sentences from the target document(s). If the summaries are query-biased then effective techniques to measure query-sentence similarities are needed [16]. Information Extraction methods involve often some sentence retrieval algorithm to support their processes [12]. Sentence retrieval mechanisms have also been found important in Machine Translation [6].

Given a set of documents, our work focuses on a retrieval task based on selecting sentences relevant to a given information need, which is expressed as a textual query. This sentence retrieval problem is delimited to work with documents highly related to the query. This simulates a working environment in which an initial document retrieval is run and, next, the top ranked documents are input to a sentence retrieval module that filters out the irrelevant sentences and supplies the user with a rank of sentences. As argued in [18], a sentence retrieval interface of this kind would be very valuable, especially for searches in which the user does not have a clear idea about the topics involved and the sentences supplied can help her/him to clarify the purpose of the search.

Query expansion strategies, which have played a major role in document retrieval, are not sufficiently tested for sentence retrieval problems. Although some works have reported improvements using classical expansion techniques via pseudo-relevance feedback [8], there are no comparisons available testing extensively different term selection methods and studying the effect of the number of sentences and terms used for expansion. Expansion strategies developed for document retrieval might be ineffective for sentence retrieval because the number of matching terms is much smaller and, thus, performance might be harmed. Due to the importance of query expansion in sentence retrieval, we feel strongly that a complete study on this subject is required. The vocabulary mismatch problem is a severe obstacle to yield effective retrieval at the sentence level and the role of query expansion as an alleviation tool needs to be carefully analyzed. Furthermore, there are no comparative results between expansion before sentence retrieval and expansion after sentence retrieval. Expansion before sentence retrieval has been par-

SIGIR 2008 Workshop on Focused Retrieval.
July 24, 2008, Singapore.

ticularly neglected. Since we start from a set of top ranked documents, it makes sense to study blind feedback methods working directly with the initial ranking of documents and compare them with regular pseudo-relevance feedback applied after running a first sentence retrieval process. Note also that this has important implications for efficiency that should not be disregarded.

Our study will be primarily focused on two standard automatic expansion methods that have worked well in document retrieval problems: pseudo-relevance feedback (PRF) [4] and Local Context Analysis (LCA) [19]. These techniques are general enough to be applied across different domains and collections. Although some works have managed to get effective expansion with linguistic resources, we are concerned here only with purely statistical methods, which are simpler and applicable under very distinct scenarios.

The rest of the paper is organized as follows. Section 2 reviews some papers related to our research. Section 3 presents the sentence retrieval method and the expansion techniques tested. The experiments are reported and analyzed in section 4. The paper ends with some conclusions.

2. RELATED WORK

Sentence retrieval is a challenging area. Many researchers have proposed different solutions based on a wide range of models and techniques such as query expansion (either via pseudo-relevance feedback or with the aid of a lexical resource), part-of-speech tagging, clustering of sentences, named entities, supervised learning, and language modeling. Despite the variety of the approaches investigated, simple adaptations of regular tf/idf measures (sometimes aided with some form of pseudo-relevance feedback) can be labeled as state of the art sentence retrieval methods [2, 9].

Many studies have examined the use of expanded queries either via pseudo-relevance feedback [5] or with the assistance of a terminological resource, such as Wordnet [20]. The effect of pseudo-relevance feedback is known to be very sensitive to the quality of the initial ranks. Motivated by this, some researchers have applied selective feedback [1], which is more stable but requires training data. In [11] some experiments investigating the effects of pseudo-relevance feedback on sentence retrieval were reported. Query expansion produced negative results but a single expansion technique, based on Relevance Models, was tested. On the other hand, expansion with synonyms or related terms from a lexical resource is problematic because noisy terms can be easily introduced into the new query. Moreover, a large terminological resource, with good coverage, is not always available. As a matter of fact, lexical expansion is usually equal or inferior to purely statistical expansion methods in sentence retrieval [7, 15, 14].

Expansion approaches based on co-occurrence data have been also proposed [20] but there is not much evidence that these approaches can outperform the standard pseudo-feedback methods.

Rather than expanding queries with new terms, other studies have focused on improving the matching process by analyzing carefully the nature of the sentence components. For example, in [9], patterns such as phrases, combinations of query terms and named entities were identified into sentences and the sentence retrieval process was driven by such artifacts. Although this technique was very effective for detecting redundant sentences, it was not significantly better than a regular tf/idf baseline for finding relevant sentences.

In [10], an effective sentence retrieval method, based on extracting highly frequent terms from top ranked documents, was designed. This method actually represents a form to exploit the information from top retrieved documents before sentence retrieval. It was successfully compared against query expansion using pseudo-relevance feedback from top retrieved sentences (i.e. expansion after sentence retrieval). Nevertheless, expansion before sentence retrieval (i.e. expanding directly from the top retrieved documents) was not properly tested. For instance, sophisticated expansion techniques, such as LCA, were not considered in the experimental design.

There is therefore the general feeling in the sentence retrieval community that some form of expansion is needed to achieve reasonably good performance. However, expansion methods have not been adequately compared and, actually, we can find in the literature conflicting outcomes depending on the collection, baseline method tested, etc. In the present work we aim to clarify the role of expansion strategies in sentence retrieval by testing some standard methods against three different datasets and applying a very competitive baseline.

In the literature of sentence retrieval, the peculiarities of the sentence retrieval task are often ignored. Most expansion studies do not make full use of the information available but simply apply expansion methods that worked well in document retrieval. We argue that the ranked set of documents contains valuable information on the importance of terms that should not be disregarded. In this respect, we believe that it is important to check the effectiveness of query expansion methods when applied before sentence retrieval (i.e. working directly on the top retrieved documents available). There are at least two reasons that support this claim. First, sentence retrieval is very sensitive to the quality of the query and, hence, we might be safer working on the initial set of documents rather than on a subsequent ranking of sentences. Second, it would avoid retrieving an initial ranking of sentences and therefore would bring about a benefit in terms of efficiency.

3. SENTENCE RETRIEVAL METHOD

To study properly different query expansion strategies we need first to decide which sentence retrieval method is appropriate for our purposes. Since we want to evaluate the ability of expansion techniques to improve the state-of-the-art in sentence retrieval, we have to set a competitive sentence retrieval technique. In [2], the results of some sentence retrieval experiments are discussed. A simple vector space retrieval technique is shown to perform at least as well as any other method and, actually, its performance is the most robust. This method, which we will refer to as tf/isf[1], applies a weighting scheme that is a variant of tf/idf applied at the sentence level. Although other effective methods, such

[1]isf stands for inverse sentence frequency

as those based on clusters of sentences, can be found in the literature [7, 15, 14], we skip them deliberately because the tf/isf method is simpler and we therefore avoid possible biases and complications coming from evolved approaches (e.g. the effect of the quality of the clusters). We believe strongly that the simplicity of this method is a good feature, making the results presented here potentially applicable in very different scenarios. The relevance of a sentence s given a query q is estimated in [2] as:

$$tf\_isf(s,q) = \sum_{t \in q} log(tf_{t,q} + 1) log(tf_{t,s} + 1) log(\frac{n+1}{0.5 + sf_t}) \quad (1)$$

where sf_t is the number of sentences in which t appears, n is the number of sentences in the collection and $tf_{t,q}$ ($tf_{t,s}$) is the number of occurrences of t in q (s).

To further check that tf/isf was competitive we designed some preliminary experiments whose results are reported in section 4.1. This included experiments using alternative sentence retrieval methods (OKAPI BM25 and Language Modeling with KLD), different combinations of the pre-processing strategies and even additional tests using idf statistics (instead of isf). This evaluation demonstrated clearly that tf/isf is a consistent sentence retrieval method whose performance is comparable or superior to the best performance attainable by other effective methods.

3.1 Expansion after sentence retrieval

By query expansion *after* sentence retrieval (ASR) we refer to the regular pseudo-relevance feedback process adapted to the sentence retrieval case. First, the query is run against the sentences in the top retrieved documents and, next, the top retrieved sentences are used to mine expansion terms. Two main strategies are considered to select new terms: PRF and LCA.

Pseudo-relevance feedback (also called local or blind feedback) is a traditional concept in IR [3], which basically consists of selecting the terms with more counts in the top retrieved sentences. Although it did not work well with small (pre-TREC) collections, its merits for large-scale document retrieval have been apparent in many TREC experiments [17]. Nevertheless, the effects this method has on sentence retrieval have not been studied in detail. Actually, some papers have reported improvements after expansion via pseudo-relevance feedback but other studies are sceptical about local feedback improving sentence retrieval [11]. We therefore expect that the experiments reported here help to shed light on this issue. Note also that there are some parameters needed for success, such as the number of top sentences and the number of expansion terms. Sentences are very small pieces of text and retrieval performance may be very sensitive to the parameter configuration here.

LCA is a successful expansion method proposed by Xu and Croft [19]. It has been adopted by other research groups in several large-scale experiments in document retrieval [17]. Nevertheless, the effects of LCA in sentence retrieval are barely discussed in the literature and there are no experimental results available comparing LCA and local feedback. The main motivation to propose LCA was that local feedback fails if there is a large number of non-relevant items in the top ranked set. The LCA method tries to be less erratic and is designed to work on document passages. We take here an instance of the LCA proposal where passages are simply document sentences. The main hypothesis behind LCA is that common terms from relevant documents (sentences, in our case) will tend to co-occur with query terms within the top-ranked documents (sentences). In this way, a term selection metric is defined, yielding an expansion method that is more robust than local feedback. Although LCA works for *concept* selection, where concepts can be single terms or phrases, we are only concerned here with selecting single terms for expansion. Let us consider a query q, whose query terms are $qt_1, \cdots, qt_m$, and a set of top ranked sentences $S = \{s_1, s_2, \cdots, s_n\}$. The terms appearing in S are ranked according to the formula:

$$\begin{aligned} f(t,q) &= \prod_{qt_i \in q} (\delta + co\_degree(t, qt_i))^{idf(qt_i)} \\ co\_degree(t, qt_i) &= log_{10}(1 + co(t, qt_i)) \cdot idf(qt_i)/log_{10}(n) \\ co(t, qt_i) &= \sum_{s_j \in S} tf(t, s_j) \cdot tf(qt_i, s_j) \\ idf(t) &= min(1.0, log_{10}(N/N_t)/5.0) \end{aligned}$$

where N is number of sentences in the collection, N_t is number of sentences in the collection containing t, $tf(w, s_j)$ is the number of occurrences of w in sentence s_j and δ is a constant set to 0.1 to avoid zero value. This term ranking function is a variant of the regular tf/idf measure utilized popularly in IR. Most often preferred terms will be those rare terms (idf effect) that co-occur frequently with many query terms.

Given this measure, the terms in the retrieved sentences can be ordered in decreasing order of $f(t,q)$ and the top ranked terms are selected to expand the query.

For simplicity, we do not consider here any parameterized re-weighting strategy (e.g. based on Rocchio's formula). In both methods (pseudo-relevance feedback and LCA), the selected terms are simply incorporated as new terms in the query. Note that this involves expansion (new terms that were not present in the original query) but also basic re-weighting (the query term frequency is increased for terms belonging to the old query that are also selected in the expansion phase).

3.2 Expansion before sentence retrieval

One strategy that has not received much attention is to run query expansion before retrieving any sentence (BSR). This alternative was not explored in the past but it could become very valuable. First, for efficiency reasons: we can skip the initial sentence retrieval process (no sentence retrieval is required for doing expansion). Second, query expansion may be more robust if we work directly from the top ranked documents. Observe that poor queries will likely introduce a great deal of noise if we use them again to retrieve some sentences to feed the term selection module. The initial

ranking of documents is arguably weak for such queries but, still, a second usage of the original topic for query expansion purposes might be not advisable. It is therefore interesting to evaluate empirically these issues and compare expansion BSR and expansion ASR.

Some experiments were designed to evaluate expansion BSR. The term selection methods were the same as those explained in the previous section but the sentences used to mine the expansion terms are taken directly from the initial ranking of documents available for the task. More specifically, the top X documents (X is a parameter) are used for term mining. To maintain consistency with the ASR experiments, term selection works also at the sentence level. The BSR-version of LCA extracts new terms analyzing the co-occurrences in the sentences of the top X documents. Similarly, expansion BSR with pseudo-relevance feedback incorporates into the query the terms with more counts in the sentences of the top X documents. However, note that there is no sentence retrieval here (e.g. if $X = 1$ then all sentences from the top document are considered in the term selection process).

3.3 Complexity issues

Expansion ASR introduces an important time penalty because it requires a sentence retrieval process for term selection. In contrast, expansion BSR works directly from the sentences in the top retrieved documents. This is a considerable saving.

Given a set of sentences (either a set of sentences ranked in decreasing order of similarity to a given query -expansion ASR- or a set of sentences appearing in top ranked documents -expansion BSR-), it is interesting to compare the steps needed to compute the ranks of terms with pseudo-relevance feedback or LCA. Pseudo-relevance feedback simply requires to traverse the sentences and accumulate the term counts in a proper data structure (e.g. a term-count structure ordered by count). LCA requires also to go on every sentence and accumulate the $co(t, q_i)$ counts (for each q_i). The time complexity of this process across retrieved sentences is equivalent to the time complexity needed by pseudo-relevance feedback (although the space complexity is higher with LCA because we need to store independent statistics for each query term). Anyway, LCA incorporates an additional time penalty to compute the final $f(t, q)$ values (product across query terms). This cost, which is linear with respect to the number of query terms, could be assumed to be negligible, especially if queries are short.

4. EXPERIMENTS

We designed a complete pool of experiments to test the expansion configurations. The experiments were run against three different collections of data (the ones supplied in the context of the TREC-2002, TREC-2003 and TREC-2004 novelty tracks [7, 15, 14]). There are no newer TREC collections suitable for our experiments because we need relevance judgments at the sentence level. This sort of judgments is only available in the novelty track, whose last edition took place in 2004. The novelty track data was constructed as follows. Every year there were 50 topics available. In TREC-2002, the topics were taken from TRECs 6, 7 and 8[2]. In 2003 and 2004, the topics were created by assessors designated specifically for the task [15, 14] (topics N1-N100). For each topic, a rank of documents was obtained by NIST using an effective retrieval engine. In 2002 and 2003 the task aimed at finding relevant sentences in relevant documents and, therefore, the ranks included only relevant documents (i.e. given a topic the set of relevant documents to the topic were collected and ranked using a document retrieval engine). On the contrary, the TREC-2004 ranks contained also non-relevant documents (i.e. the initial search for documents was done against a regular document base, with relevant and non-relevant documents). Note that this means that the non-relevant documents are close matches to the relevant documents, and not random non-relevant documents [14]. In any case, the ranks of documents contained at most 25 relevant documents for each query. The documents were segmented into sentences, the participants were given these ranks of sentence-tagged documents and they were asked to locate the relevant sentences. The relevance judgments in this task are complete because the assessors reviewed carefully the ranked documents and marked every sentence as relevant or non-relevant to the topic. In TREC-2002, very few sentences were judged as relevant (approximately 2% of the sentences in the documents). In TREC-2003 and TREC-2004 the average percentage of relevant sentences was much higher than in 2002 (approximately 40% in 2003 and 20% in 2004).

We consider here two different evaluation measures: the F measure, which was the official measure utilized in the TREC novelty tracks, and precision at ten sentences retrieved (P@10). The F measure is meaningful even when the number of relevant sentences varies widely across topics [7]. The F values reported here are obtained by retrieving 5% of the sentences in TREC 2002, and 50% of the sentences in TREC 2003 and TREC 2004. These thresholds, which have been applied in the past, are reasonable given the amount of relevant sentences in every collection. Additionally, P@10 ratios are included in our reports. P@10 is important in many applications, such as web sentence retrieval [18], which require a good distribution of relevant material in the top rank positions.

We focus our interest on short queries (constructed from the title tags of the TREC topics) because handling properly this type of queries is challenging in sentence retrieval. These queries are good candidates for expansion because they are often ambiguous.

4.1 Evaluating the baseline

To ensure that the baseline (tf/isf, eq. 1) is capable of yielding state of the art performance, we ran some preliminary experiments comparing it against Okapi BM25 [13] and a Language Modeling approach based on Kullback-Leibler Divergence (KLD) as described in [8] (with Dirichlet smoothing). The performance of BM25 is influenced by some parameters: $k1$ controls the term frequency effect, b controls a length-based correction and $k3$ is related to query term frequency. We tested exhaustively different parameter con-

[2]The complete list of topics chosen for the novelty track can be found in [7]

	TREC-2002		
	tf/isf	BM25 (best run) $k1 = .4, b = 0, k3 = 1$	KLD (best run) $\mu = 3000$
P@10	.19	.19	.16
F	.19	.19	.17*
	TREC-2003		
	tf/isf	BM25 (best run) $k1 = .6, b = 0, k3 = 1$	KLD (best run) $\mu = 1000$
P@10	.74	.76	.73
F	.51	.51	.50*
	TREC-2004		
	tf/isf	BM25 (best run) $k1 = .2, b = 0, k3 = 1$	KLD (best run) $\mu = 500$
P@10	.43	.44	.41
F	.37	.37	.37

Table 1: Comparing different sentence retrieval baselines: tf/isf, BM25 and KLD (with Dirichlet Smoothing).

figurations ($k1$ between 0 and 2 in steps of 0.2, b between 0 and 1 in steps of 0.1 and different values of $k3$ between 1 and 1000). Similarly, we experimented with the KLD model for different values of the μ constant, which determines the amount of smoothing applied ($\mu = 10$, 100, 500, $1k$, $3k$, $5k$). Results are reported in Table 1. A run marked with an asterisk means that the difference in performance between the run and tf/isf is statistically significant[3]. In all collections, there was no statistically significant difference between the tf/isf run and the best BM25 run. We also observed that BM25 was very sensitive to the parameter setting (many BM25 runs performed significantly worse than tf/isf). On the other hand, KLD was inferior to both tf/isf and BM25. These results reinforced previous findings about the robustness of the tf/isf method [2, 9] and demonstrated that this method is a very solid baseline. Note also that tf/isf does not have any free parameter whereas the results reported for BM25 and KLD are the best ones obtained across the configurations tested.

We also experimented with different combinations of the standard preprocessing strategies (stopwords vs no stopwords, stemming vs no stemming). Although there was not much overall difference, the runs with stopword processing and no stemming were slightly more consistent.

The tf/isf method takes the isf statistics from the sentences in the documents available for the task (which is a small set of sentences). A term that is very common within the retrieved documents would therefore receive a low isf weight. This might be problematic because content-bearing terms that are frequent in a given set of documents are assigned small weights. We were therefore wondering whether better performance might be obtained using data from a larger collection. To check this, we indexed a large collection of documents (the collection used in the TREC-8 adhoc experiments) and ran some experiments with regular idf statistics obtained from this index (i.e. in eq. 1 sf_t was replaced by n_t, which is the document frequency of t in TREC-8). The original tf/isf method computed at the sentence level over the small document base was slightly superior. It appears that the small index of sentences is good enough for sentence retrieval (at least for these short queries). We therefore set the basic sentence retrieval method to be the original tf/isf approach with stopword and no stemming.

Note that we use short queries, while the groups participating in the TREC novelty tracks were allowed to use the whole topic. This means that the results presented here are not comparable to any of the results reported in the novelty tracks. Actually, we expect that the results obtained here are worse than the ones achieved in TREC because of our experimental conditions. Nevertheless, short queries are the rule rather than the exception in many applications and it is therefore important to study in depth the sentence retrieval performance with such queries. Moreover, query expansion methods are especially important when the user supplies few search terms.

4.2 Evaluating query expansion strategies

Let us now pay attention to the effects of query expansion on sentence retrieval performance. With expansion ASR, we first ran the tf/isf sentence retrieval method on the ranked set of documents associated to each query. This produced a ranked set of sentences from which some expansion terms were selected using PRF or LCA. These new terms were included into the query and the tf/isf sentence retrieval model was run again with the expanded query. We tested different configurations of the number of expansion terms (5, 10, 20 and 50) and the number of top retrieved sentences in which terms are selected (5, 10, 25, 50 and 100). On the other hand, expansion BSR selects terms directly from the ranked set of documents. We planned experiments using the top 1, 5, 10, 15 or 25 documents with varying number of expansion terms.

The experimental results are reported in Tables 2 (expansion ASR - P@10), 3 (expansion BSR - P@10), 4 (expansion ASR - F measure) and 5 (expansion BSR - F measure). The tables include also the performance of the baseline tf/isf with no expansion (underlined after the collection's name). For each collection and type of expansion the best parameter configurations are marked in bold. Expansion runs whose improvement over the baseline is statistically significant are marked with an asterisk.

First of all, it is interesting to observe the effect of expansion on the TREC-2002 collection. There are some expansion configurations that show P@10 and F ratios that are higher than the baseline's ratios. Anyway, nearly all improvements are not statistically significant. Observe that this collection contains very few relevant sentences ($\approx 2\%$) and, therefore, any expansion strategy is likely incorporating unrelated terms into the new queries. PRF is particularly problematic here because it often performs worse than the baseline (32 out of the 80 TREC-2002 PRF expansion runs perform worse than the baseline). In contrast, LCA does not improve significantly over the baseline but, at least, there are fewer LCA runs yielding performance that is poorer than the baseline's performance (14 runs out of 80).

On the other hand, most expansion methods produce statistical significant improvements in TREC-2003 and TREC-

[3]Along this work, we applied two different significance tests, the t-test and the Wilcoxon test, and we show only an asterisk when both tests agree on the significance of the difference (95% confidence level).

	PRF					LCA				
# exp terms	# top sentences 5	10	25	50	100	# top sentences 5	10	25	50	100
TREC-2002 (basel: .19)										
5	.19	.19	.22	**.24***	.23	.20	.19	.19	.21	.21
10	.21	.20	.21	**.24**	**.24***	.19	.18	.19	.21	**.23**
20	.19	.17	.20	.21	.23	.18	.16	.19	.22	.22
50	.19	.20	.20	.20	.22	.18	.18	.19	.21	.22
TREC-2003 (basel: .74)										
5	.78*	.78*	**.81***	.79	.79*	.75	.75	.75	.75	.75
10	.78*	.79*	.80*	**.81***	.80*	.79*	.78*	.80*	.81*	.80
20	.80*	.78	.80*	.80	.79	.80*	.78*	.80*	**.82***	.79
50	.78*	.77	.79	.75	.76	.79*	.77	.79*	.79	.81*
TREC-2004 (basel: .43)										
5	.46	.48*	.49*	.50*	.48*	.45	.47	.51*	.51*	.47
10	.47	.48*	.51*	.54*	.54*	.47	.50*	.49*	.49*	.50*
20	.47	.49*	.50*	**.55***	**.55***	.46	.47	.48	.49*	.49*
50	.44	.46	.50*	.52*	.54*	.45	.46	.49*	.50*	**.53***

Table 2: Expansion ASR - Precision at 10 sentences

	PRF					LCA				
# exp terms	# docs 1	5	10	15	25	# docs 1	5	10	15	25
TREC-2002 (basel: .19)										
5	.19	.20	**.22**	.19	.19	.20	.22	.21	.21	.21
10	.16	.19	.21	.21	.17	.20	.22	.23	.23	.23
20	.16	**.22**	.19	.19	.17	.18	.22	.22	.21	.23
50	.17	**.22**	.20	.18	.17	.17	**.24**	.23	.22	.22
TREC-2003 (basel: .74)										
5	.75	.77	.76	.75	.77	.74	.77	.78	.77	.74
10	.74	.78	.76	.77	**.79**	.72	.79	.78	.76	.79
20	.75	**.79**	.77	.77	.76	.71	.77	.79	**.81***	.80
50	.71	.75	**.79**	.78	.76	.71	.78	.79	**.81***	.77
TREC-2004 (basel: .43)										
5	.42	.49	.47	.50*	.48*	.42	.46	.46	.47	.49
10	.39	.50*	.50*	.49*	.51*	.43	.47	.48	.50*	.48
20	.38	.49	.50*	.50*	.54*	.43	.48	.49	.49	.50*
50	.37	.46	.51*	.54*	**.55***	.40	.43	.49	.49	**.52***

Table 3: Expansion BSR - Precision at 10 sentences

2004 (for both P@10 and F). These results show that statistical query expansion is beneficial in sentence retrieval provided that the amount of relevance sentences in the ranked set of documents is not extremely low.

Next, we analyze the trends with respect to the number of expansion terms and the number of top sentences/documents.

Expansion ASR, P@10 (Table 2). The standard PRF expansion tends to achieve the highest P@10 performance when a few expansion terms are selected from many sentences. A safe configuration would be 10 expansion terms selected from 50 or 100 sentences. With LCA-based expansion, the ideal number of sentences is also high but this expansion method tolerates slightly better expansions with more terms.

Expansion BSR, P@10 (Table 3). When expanding queries before sentence retrieval, PRF looks much more sensitive to the parameter setting. It is quite difficult to identify a good configuration because the optimal performance is found at very different places depending on the collection. Only in TREC-2004 the improvements over the baseline are statistical significant. On the other hand, LCA looks less erratic. A high number of terms (50) extracted from a large number of documents (15-25) seems to be a good configuration.

	PRF					LCA				
# exp terms	# top sentences 5	10	25	50	100	# top sentences 5	10	25	50	100
TREC-2002 (5% sens ret.) (basel: .19)										
5	.19	.19	.19	.19	.19	.20	.19	.19	.19	.19
10	.19	.18	.19	.20	.20	.19	.18	.18	.19	**.20**
20	.18	.18	.19	.20	**.21**	.18	.18	.18	**.20**	**.20**
50	.18	.18	.19	.20	**.21**	.18	.18	.19	**.20**	**.20**
TREC-2003 (50% sens ret.) (basel: .51)										
5	.53	.54*	.54*	.55*	.55*	.52	.52*	.53*	.53*	.53*
10	.55*	.55*	.55*	**.57***	.56*	.53*	.53*	.55*	.55*	.55*
20	.55*	.56*	.56*	.56*	**.57***	.55*	.55*	.56*	.56*	.56*
50	.56*	.56*	**.57***	**.57***	**.57***	.56*	.56*	.56*	**.57***	**.57***
TREC-2004 (50% sens ret.) (basel: .37)										
5	.38	.38	.38	.38	.39*	.37	.37	.37	.38	.38
10	.38	.38	.39*	.39*	.39*	.38	.38*	.38*	.39*	.39*
20	.39*	.39*	.39*	.40*	.40*	.38*	.39*	.39*	.39*	.39*
50	.39*	.39*	.40*	**.41***	.40*	.39*	.39*	**.40***	**.40***	**.40***

Table 4: Expansion ASR - F measure

	PRF					LCA				
# exp terms	# docs 1	5	10	15	25	# docs 1	5	10	15	25
TREC-2002 (5% sens ret.) (basel: .19)										
5	.17	.18	.18	.16	.17	.19	.19	.20	.20	.19
10	.17	**.19**	.18	.17	.16	.18	.20	.20	.20	.20
20	.17	.18	.18	.17	.16	.18	.19	**.21**	.20	.20
50	.16	.18	.18	.17	.15	.19	.20	**.21**	**.21**	**.21**
TREC-2003 (50% sens ret.) (basel: .51)										
5	.55*	.55*	.55	.55*	.55*	.52*	.53*	.53*	.53*	.54*
10	.55*	**.56***	.55	**.56**	**.56***	.52*	.55*	.54*	.54*	.56*
20	.56*	.55*	.55	**.56**	**.56***	.55*	.55*	.56*	.56*	**.57***
50	.55	**.56***	**.56***	**.56***	**.56***	.56*	.56*	**.57***	**.57***	**.57***
TREC-2004 (50% sens ret.) (basel: .37)										
5	.37	.38	.38	.38	.38	.37	.38	.38	.38	.38
10	.38	.38	.38	.38	.38	.37	.38	.39*	.39*	.39*
20	.38	.38	.39	.39	.39*	.38	.39*	.39*	.39*	.39*
50	.38	.39	.39*	**.40***	**.40***	.38	.39*	.40*	.40*	**.41***

Table 5: Expansion BSR - F measure

Expansion ASR, F-measure (Table 4). With both expansion methods, PRF and LCA, the highest performance tends to be found when a large number of expansion terms are selected from a large number of top sentences. P@10 is a high precision measure but the F measure is influenced by both precision and recall. This explains why the optimal F performance is found with expansions involving many new terms, whilst the optimal P@10 performance is achieved usually with fewer expansion terms.

Expansion BSR, F-measure (Table 5). With LCA, there is also a clear tendency to prefer many expansion terms extracted from many documents. However, with PRF, the optimal configuration varies significantly depending on the collection. These results agree with those found for P@10 with BSR.

Given this report, it is clear that LCA performs the best with many expansion terms extracted either from many sentences (ASR) or from many documents (BSR). PRF is much more erratic and its optimal expansion configuration is much more difficult to assess.

Let us now analyze the best and average performance attainable by each expansion method. For a clearer picture

of the experimental outcome, these results are summarized in Table 6. The difference between the best ASR and BSR runs has been tested for statistical significance and the BSR run is marked with the symbol † when the difference between the run and the respective ASR run is significant. In terms of P@10, there is no significant difference between the best runs. This means that any configuration (ASR/BSR + PRF/LCA) can lead to optimal performance provided that the parameters (number of expansion terms and number of top sentences/documents) are set adequately. Looking at the average P@10 values, we found some interesting trends. With expansion ASR, PRF is more solid than LCA. On the contrary, with expansion BSR, LCA tends to be more reliable (especially when the conditions are difficult -few relevant sentences-). This makes sense because the sentences feeding the ASR term selection module are potentially closer to the query than the sentences feeding the BSR term selection module. Recall that expansion ASR runs an initial sentence retrieval from the query and the retrieved sentences are used for term selection purposes. In contrast, expansion BSR works directly with the initial ranked set of documents, where the on-topic sentences might be scattered across the documents. This means that a rough term selection metric (such as local feedback) is good enough with expansion ASR but it is less consistent when there is not an initial sentence retrieval process.

In terms of the F measure, the results are basically the same as the ones found with P@10. PRF tends to work better with expansion ASR while LCA tends to be more solid with expansion BSR. In two collections the best run of PRF with expansion ASR performs statistically significantly better than the best run of PRF with expansion BSR. It is interesting to note that the single collection where the difference is not significant is TREC-2002, where there are few relevant sentences. This makes sense because expansion ASR is very sensitive to the quality of the initial sentence retrieval process. If these ranked sentences contain many non-relevant items then expansion ASR will hardly improve on expansion BSR.

In terms of effectiveness, expansion ASR with PRF and expansion BSR with LCA are the most robust expansion methods for sentence retrieval. Both approaches lead to good P@10 and F performance ratios. Since expansion BSR is less expensive than expansion ASR (because we do not need an initial sentence retrieval process), expansion BSR with LCA looks the most suitable choice. One can rightly argue that LCA is more costly than PRF but, as argued in section 3.3, the additional complexity requirements are acceptable. This means that we can achieve state-of-the-art sentence retrieval performance with significant savings in terms of efficiency. This is a novel result because the studies conducted in the literature have been mostly focused on the standard expansion methods (ASR). Furthermore, if the aim of the retrieval application is to retrieve ten good sentences (i.e. recall is not a major issue) then expansion BSR with PRF is a good choice. As shown in Table 6, this retrieval technique, which is the most efficient method, does not perform significantly worse than the other expansion methods (in terms of P@10).

		TREC-2002			
		ASR	BSR	ASR	BSR
		best		*avg*	
P@10	PRF	.24	.22	.21	.18
(basel: .19)	LCA	.23	.24	.20	.22
F	PRF	.21	.19	.19	.17
(basel: .19)	LCA	.20	.21	.19	.20
		TREC-2003			
P@10	PRF	.81	.79	.79	.76
(basel: .74)	LCA	.82	.81	.78	.77
F	PRF	.57	.56†	.56	.55
(basel: .51)	LCA	.57	.57	.55	.55
		TREC-2004			
P@10	PRF	.55	.55	.50	.48
(basel: .43)	LCA	.53	.52	.48	.47
F	PRF	.41	.40†	.39	.38
(basel: .37)	LCA	.40	.41	.39	.39

Table 6: Comparing the best and average performance of expansion ASR and expansion BSR with PRF and LCA

5. CONCLUSIONS

In this paper we have presented a thorough study on the effects of query expansion strategies for sentence retrieval. We have worked with an standard sentence retrieval method, proved that it is competitive against other robust techniques and supplied a complete study of query expansion under this framework.

The results of our study can be summarized as follows. In terms of effectiveness, expansion ASR with PRF and expansion BSR with LCA are the most robust expansion methods for sentence retrieval. Both approaches lead to good P@10 and F performance ratios. Since expansion BSR is less expensive than expansion ASR (because we do not need an initial sentence retrieval process), expansion BSR with LCA looks to be the most suitable choice. This means that we can achieve state-of-the-art sentence retrieval performance with significant savings in terms of efficiency. This is a novel result because the studies conducted in the literature have been mostly focused on the standard expansion methods (ASR).

Regarding the number of expansion terms and the number of top documents/sentences from which terms are mined, we found that the more documents/sentences fed into the expansion modules the better performance. This is not very surprising. On the other hand, LCA shows a slight tendency to achieve its highest performance with expansions involving many terms while PRF is more erratic with respect to the ideal number of expansion terms. In general, PRF is very sensitive to the parameter setting. Although the top performance attainable by PRF tends to be similar to LCA's top performance, the parameter settings are more problematic with PRF.

Summing up, although some past studies have been skeptical on the role of query expansion for sentence retrieval, our report shows that it is a consistent technique to improve sentence retrieval performance provided that the retrieved documents contain a reasonable amount of relevant sentences. The two methods tested, PRF and LCA, can produce significant benefits when parameters are set appro-

priately. Even with an extremely low population of relevant sentences (TREC-2002), a proper query expansion configuration (e.g. BSR+LCA for high precision purposes and ASR+PRF otherwise) hardly damages performance.

ACKNOWLEDGMENTS

I thank the support obtained from projects TIN2005-08521-C02-01 (*Ministerio de Educación y Ciencia*) and PGIDIT07-SIN005206PR (*Xunta de Galicia*).

6. REFERENCES

[1] N. Abdul-Jaleel, J. Allan, B. Croft, F. Diaz, L. Larkey, X. Li, M. Smucker, and C. Wade. UMass at TREC 2004: Novelty and hard. In *Proceedings of the 13th Text Retrieval Conference (TREC 2004)*, 2004.

[2] J. Allan, C. Wade, and A. Bolivar. Retrieval and novelty detection at the sentence level. In *Proc. SIGIR-03, the 26th ACM Conference on Research and Development in Information Retrieval*, pages 314–321, Toronto, Canada, 2003. ACM press.

[3] R. Attar and A. Fraenkel. Local feedback in full-text retrieval systems. *Journal of the Association for Computing Machinery*, 24(3):397–417, 1977.

[4] C. Buckley, A. Singhal, M. Mitra, and G. Salton. New retrieval approaches using SMART: TREC 4. In D.Harman, editor, *Proc. TREC-4*, pages 25–48, 1996.

[5] K. Collins-Thompson, P. Ogilvie, Y. Zhang, and J. Callan. Information filtering, novelty detection, and named-page finding. In *Proc. TREC-2002, the 11th text retrieval conference*, 2002.

[6] T. Doi, H. Yamamoto, and E. Sumita. Example-based machine translation using efficient sentence retrieval based on edit-distance. *ACM Transactions on Asian Language Information Processing*, 4(4):377–399, 2005.

[7] D. Harman. Overview of the TREC 2002 novelty track. In *Proc. TREC-2002, the 11th text retrieval conference*, 2002.

[8] L. Larkey, J. Allan, M. Connell, A. Bolivar, and C. Wade. UMass at TREC 2002:cross language and novelty tracks. In *Proc. TREC-2002, the 11th text retrieval conference*, 2002.

[9] X. Li and B. Croft. Novelty detection based on sentence level patterns. In *Proc. of CIKM-2005, The ACM Conference on Information and Knowledge Management*, 2005.

[10] D. Losada and R. T. Fernández. Highly frequent terms and sentence retrieval. In *Proc. 14th String Processing and Information Retrieval Symposium, SPIRE'07*, Santiago de Chile, October 2007.

[11] V. Murdock. *Aspects of sentence retrieval*. PhD thesis, University of Massachussetts, 2006.

[12] C. Nobata and S. Sekine. Towards automatic acquisition of patterns for information extraction. In *Proc. of International Conference of Computer Processing of Oriental Languages*, 1999.

[13] S. Robertson, S. Walker, S. Jones, M. HancockBeaulieu, and M. Gatford. Okapi at TREC-3. In D.Harman, editor, *Proc. TREC-3, the 3rd Text Retrieval Conference*, pages 109–127. NIST, 1995.

[14] I. Soboroff. Overview of the TREC 2004 novelty track. In *Proc. TREC-2004, the 13th text retrieval conference*, 2004.

[15] I. Soboroff and D. Harman. Overview of the TREC 2003 novelty track. In *Proc. TREC-2003, the 12th text retrieval conference*, 2003.

[16] A. Tombros and M. Sanderson. Advantages of query biased summaries in information retrieval. In *Proc. of SIGIR-98, the 21st ACM International Conference on Research and Development in Information Retrieval*, pages 2–10. ACM press, August 1998.

[17] E. Voorhees and D. Harman, editors. *TREC:Experiment and Evaluation in Information Retrieval*, chapter The TREC AdHoc Experiments, pages 79–97. The MIT press, 2005.

[18] R. White, J. Jose, and I. Ruthven. Using top-ranking sentences to facilitate effective information access. *Journal of the American Society for Information Science and Technology (JASIST)*, 56(10):1113–1125, 2005.

[19] J. Xu and B. Croft. Query expansion using local and global document analysis. In *Proc. SIGIR-96, the 19th ACM Conference on Research and Development in Information Retrieval*, pages 4–11, Zurich, Switzerland, July 1996.

[20] H. Zhang, H. Xu, S. Bai, B. Wang, and X. Cheng. Experiments in TREC 2004 novelty track at CAS-ICT. In *Proc. TREC-2004, the 13th text retrieval conference*, 2004.