Dealing with Syntactic Variation through a Locality-Based Approach

Jesús Vilares and Miguel A. Alonso

Departamento de Computación, Universidade da Coruña
Campus de Elviña s/n, 15071 A Coruña, Spain.
{jvilares, alonso}@udc.es
http://www.grupocole.org/

Abstract. To date, attempts for applying syntactic information in the document-based retrieval model dominant have led to little practical improvement, mainly due to the problems associated with the integration of this kind of information into the model. In this article we propose the use of a locality-based retrieval model for reranking, which deals with syntactic linguistic variation through similarity measures based on the distance between words. We study two approaches whose effectiveness has been evaluated on the CLEF corpus of Spanish documents.

1 Introduction

Syntactic processing has been applied repeatedly in the field of Information Retrieval (IR) for dealing with the syntactic variation present in natural language texts [14, 8, 11], although its use in languages other than English has not as yet been studied in depth. In order to apply these kind of techniques, it is necessary to perform some kind of parsing process, which itself requires the definition of a suitable grammar. For languages lacking advanced linguistics resources, such as wide-coverage grammars or treebanks, the application of these techniques is a real challenge. In the case of Spanish, for example, only a few IR experiments involving syntax have been performed [1, 18, 20, 19]. Even when reliable syntactic information can be extracted from texts, the issue that arises is how to integrate it into an IR system. The prevalent approaches consist of a weighted combination of multi-word terms —in the form of head-modifier pairs— and single-word terms —in the form of word stems—. Unfortunately, the use of multi-word terms has not proven to be effective enough, regardless of whether they have been obtained by means of syntactic or statistical methods, mainly due to the difficulty of solving the overweighting of complex terms with respect to simple terms [13].

In this context, pseudo-syntactic approaches based on the distance between terms arise as a practical alternative that avoids the problems listed above as a result of not needing any grammar or parser, and because the information about the occurrence of individual words can be integrated in a consistent way with the information about proximity to other terms, which in turn is often related with the existence of syntactic relations between such terms.

In this work we propose the use of a *locality-based retrieval model*, based on a similarity measure computed as a function of the distance between terms, as a complement to classic IR techniques based on indexing single-word terms, with the aim of increasing the precision of the documents retrieved by the system in the case of Spanish.

The rest of the article is organized as follows. Section 2 introduces the locality-based retrieval model and our first approach for integrating it into our system; the experimental results of this first proposal are shown in Section 3. A second approach, based on data fusion, is described in Section 4, and its results are discussed in Section 5. Finally, our conclusions and future work are presented in Section 6.

2 Locality-Based IR

2.1 The Retrieval Model

In the *document-based* retrieval model prevalent nowadays, an IR system retrieves a list of documents ranked according to their degree of relevance with respect to the information need of the user. In contrast, a *locality-based* IR system goes one step further, and looks for the concrete *locations* in the documents which are relevant to such a need. *Passage retrieval* [10] could be considered as an intermediate point between these two models, since its aim is to retrieve portions of documents (called *passages*) relevant to the user. However, passage retrieval is closer to document-based than to locality-based retrieval: once the original documents have been split into passages they are ranked using traditional similarity measures. In this case, the main difficulty comes from specifying what a passage is, including considerations about size and overlapping factors, and how they can be identified.

In contrast, the locality-based model considers the collection to be indexed not as a set of documents, but as a sequence of words where each occurrence of a query term has an influence on the surrounding terms. Such influences are additive, thus, the contributions of different occurrences of query terms are summed, yielding a similarity measure. As a result, those areas of the texts with a higher density of query terms, or with important query terms, show peaks in the resulting graph, highlighting those positions of the text which are potentially relevant with respect to the query. A graphical representation of this process is shown in Fig. 1. It is worth noting that relevant portions are identified without the need to perform any kind of splitting in the documents, as is done in passage retrieval.

Next, we describe the original proposal of de Kretser and Moffat for the locality-based model [5, 6].

2.2 Computing the Similarity Measure

In the locality-based model the similarity measure only needs to be computed for those positions of the text in which query terms occur, a characteristic which

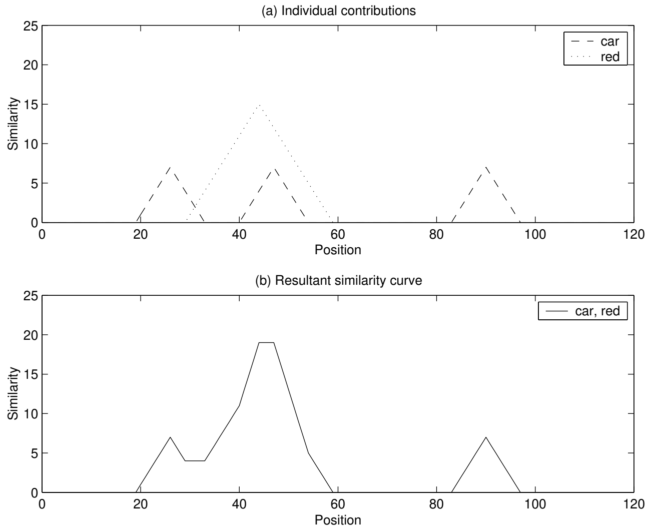

Fig. 1. Computing the similarity measure in a locality-based model: (a) positions where query terms occur and their regions of influence; (b) the resultant similarity curve

makes its application possible in practical environments due to its computational cost being relatively low.

The contribution to the similarity graph of a given query term is determined by a *similarity contribution function* c_t defined according to the following parameters [5]:

- The *shape* of the function, which is the same for all terms.
- The maximum *height* h_t of the function, which occurs in the position of the query term.
- The *spread* s_t of the function, that is, the scope of its influence.
- The distance, in words, with respect to other surrounding words, $d = |x - l|$, where l is the position of the query term and x is the position of the word in the text where we want to compute the similarity score.

Several function shapes are described in [5], but we only show here those with which we obtained better results in Spanish. They are the triangle (*tri*) and the circle (*cir*) function, defined by equations 1 and 2, respectively, and whose graphical representation is shown in Fig. 2:

$$c_t(x,l) = h_t(1 - d/s_t) \; . \tag{1}$$

$$c_t(x,l) = h_t\sqrt{1 - (d/s_t)^2} \; . \tag{2}$$

with $c_t(x,l) = 0$ when $|x - l| > s_t$.

The height h_t of a query term t is defined as an inverse function of its frequency in the collection:

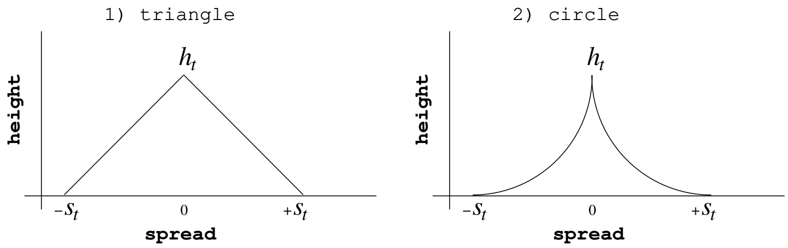

Fig. 2. Shapes of the similarity contribution function c_t

$$h_t = f_{q,t}\ log_e(N/f_t)\ . \tag{3}$$

where N is the total number of terms in the collection, f_t is the number of times term t appears in the collection, and $f_{q,t}$ is the within-query frequency of the term.

On the other hand, the spread s_t of the influence of a term t is also defined as an inverse function of its frequency in the collection, but normalized according to the average term frequency:

$$s_t = \frac{n}{N}\frac{N}{f_t} = \frac{n}{f_t}\ . \tag{4}$$

where n is the number of unique terms in the collection, that is, the size of the vocabulary.

Once these parameters have been fixed, the similarity score assigned to a location x of the document in which a term of the query Q can be found is calculated as:

$$C_Q(x) = \sum_{t \in Q} \sum_{\substack{l \in I_t \\ |l-x| \le s_t \\ term(x) \ne term(l)}} c_t(x, l)\ . \tag{5}$$

where I_t is the set of word locations at which a term t of the query Q occurs, and where $term(w)$ represents the term associated to the location w. In other words, the degree of similarity or relevance associated with a given location is the sum of all the influences exerted by the rest of query terms within whose spread the term is located, excepting other occurrences of the same term that exist at the location examined [6].

Finally, the relevance score assigned to a document D is given in function of the similarities corresponding to occurrences of query terms that this document contains. This point is discussed in detail below.

2.3 Adaptations of the Model

The locality-based model not only identifies the relevant documents but also the relevant locations they contain, allowing us to work at a more detailed level than classical IR techniques. Thus, we have opted for using this model in our experiments. Nevertheless, before doing so, the model had to be adapted to our needs, which makes our approach different from the original proposal of the model [5, 6].

The approach we have chosen for integrating distance-based similarity in our IR system consists of postprocessing the documents obtained by a document-based retrieval system. This initial set of documents is obtained through a base IR system —we name it *lem*— which employs *content-word* lemmas (nouns, adjectives and verbs) as index terms. This list of documents returned by *lem* is then processed using the locality-based model, taking the final ranking obtained using distance-based similarity as the final output to be returned to the user.

It should be pointed out that the parameters of height, h_t, and spread, s_t, employed for the reranking are calculated according to the global parameters of the collection, not according to the parameters which are local to the subset of documents returned, in order to avoid the correlation-derived problems it would introduce.[1]

Another aspect in which our approach differs from the original model is the employment of lemmatization, instead of *stemming*, for conflating queries and documents. We have made this choice due to the encouraging results previously obtained with such an approach, with respect to *stemming*, in the case of Spanish [20].

The third point of difference corresponds to the algorithm for calculating the relevance of a document, obtained from the similarity scores of its query term occurrences. Instead of the original iterative algorithm [5], our approach defines the similarity score $sim(D, Q)$ of a document D with respect to a query Q as the sum of all the similarity scores of the query term occurrences it contains:

$$sim(D, Q) = \sum_{\substack{x \in D \\ term(x) \in Q}} C_Q(x) \ . \tag{6}$$

3 Experimental Results Using Distances

Our approach has been tested using the Spanish monolingual corpus of the 2001 and 2002 CLEF editions [15], composed of 215,738 news reports provided by EFE, a Spanish news agency. The 100 queries employed, from 41 to 140, consist of three fields: a brief *title* statement, a one-sentence *description*, and a more complex *narrative* specifying the relevance assessment criteria.

As mentioned in Sect. 2.3, the initial set of documents to be reranked is obtained through the indexing of *content word lemmas* (*lem*). For this purpose, the documents were indexed with the vector-based engine SMART [3], using the `atn·ntc` weighting scheme. In order to improve the performance of the whole

[1] For example, the parameter f_t, corresponding to the number of occurrences of a term t, is the number of occurrences of t in the entire collection, not the number of occurrences of t in the set of documents to be reranked.

Table 1. Reranking based on distances

	short queries				long queries			
	stm	*lem*	*tri*	*cir*	*stm*	*lem*	*tri*	*cir*
Documents	99k	99k	99k	99k	99k	99k	99k	99k
Relevant (5548 expected)	5086	5207	5207	5207	5208	5234	5234	5234
Non-interpolated precision	.5210	.5235	.4473	.4464	.5638	.5648	.4802	.4703
Document precision	.5502	.5814	.5154	.5188	.5925	.6038	.5366	.5376
R-precision	.4952	.4978	.4438	.4453	.5316	.5335	.4574	.4490
Precision at .00 recall	.8426	.8260	**.8402**	**.8394**	.9028	.8788	.8771	.8639
Precision at .10 recall	.7294	.7431	**.7551**	**.7533**	.7910	.7989	**.8167**	**.8022**
Precision at .20 recall	.6746	.6936	.6550	.6624	.7326	.7420	.7070	.6909
Precision at .30 recall	.6135	.6380	.5764	.5806	.6763	.6887	.6066	.5996
Precision at .40 recall	.5812	.5900	.5045	.5052	.6401	.6499	.5417	.5314
Precision at .50 recall	.5470	.5520	.4496	.4515	.5975	.6058	.4894	.4819
Precision at .60 recall	.5078	.5099	.3882	.3850	.5452	.5502	.4184	.4045
Precision at .70 recall	.4518	.4498	.3360	.3340	.4816	.4816	.3654	.3547
Precision at .80 recall	.3882	.3796	.2750	.2692	.4056	.4022	.3042	.2929
Precision at .90 recall	.3044	.2923	.1933	.1917	.3356	.3150	.2023	.1944
Precision at 1.0 recall	.1897	.1756	.1031	.1014	.2054	.1918	.1062	.1000
Precision at 5 docs	.6182	.6182	.6141	.6121	.6808	.6747	.6667	.6606
Precision at 10 docs	.5717	.5758	.5596	.5596	.6182	.6202	.5929	.5869
Precision at 15 docs	.5279	.5380	.5111	.5192	.5670	.5798	.5441	.5394
Precision at 20 docs	.4965	.5071	.4803	.4818	.5338	.5556	.5081	.5056
Precision at 30 docs	.4434	.4582	.4259	.4229	.4822	.5030	.4545	.4566
Precision at 100 docs	.2935	.3016	.2691	.2696	.3119	.3171	.2811	.2812
Precision at 200 docs	.1937	.2002	.1863	.1875	.2053	.2060	.1926	.1932
Precision at 500 docs	.0945	.0981	.0964	.0964	.0981	.0985	.0979	.0982
Precision at 1000 docs	.0514	.0526	.0526	.0526	.0526	.0529	.0529	.0529

system, we have tried to obtain the best possible starting set of documents by applying pseudo-relevance feedback (blind-query expansion) adopting Rocchio's approach [16]:

$$Q_1 = \alpha Q_0 + \beta \sum_{k=1}^{n_1} \frac{R_k}{n_1} - \gamma \sum_{k=1}^{n_2} \frac{S_k}{n_2} \ . \tag{7}$$

where Q_1 is the new query vector, Q_0 is the vector of the initial query, R_k is the vector of relevant document k, S_k is the vector of non-relevant document k, n_1 is the number of relevant documents, n_2 is the number of non-relevant documents, and α, β and γ are, respectively, the parameters that control the relative contributions of the original query, relevant documents, and non-relevant documents. Our system expands the initial query automatically with the best 10 terms of the 5 top ranked documents, and using $\alpha = 1.40$, $\beta = 0.10$ and $\gamma = 0$.

It should be pointed out that the distance-based reranking process is performed according to the terms of the original query, without taking into account the terms added during the feedback. This is because there is no guarentee that these terms were syntactically related with the original query terms, since they only co-occur in the documents with such terms.

Two series of experiments have been carried out. Firstly, employing queries obtained from the title and description fields —*short queries*— and, secondly, employing queries obtained from the three fields, that is title, description and narrative —*long queries*—. It should be noticed that in the case of long queries, the terms extracted from the title field are given double relevance with respect to description and narrative, since the former summarizes the basic semantics of the query.

The results obtained are shown in Table 1. The first column of each group shows the results obtained through a standard approach based on stemming (*stm*), also using pseudo-relevance feedback; the second column contains the results of the indexing of lemmas (*lem*) before the reranking, our baseline; the two other columns show the results obtained after reranking *lem* by means of distances employing a triangle (*tri*) and circle (*cir*) function.

The performance of the system is measured using the parameters contained in each row: number of documents retrieved, number of relevant documents retrieved (5548 expected), average precision (non-interpolated) for all relevant documents (averaged over queries), average document precision for all relevant documents (averaged over relevant documents), R-precision, precision at 11 standard levels of recall, and precision at N documents retrieved. For each parameter we have marked in boldface those values where there is an improvement with respect to the baseline *lem*.

As these results show, reranking through distances has caused a general drop in performance, except for low recall levels, where results are similar or sometimes even better. We can therefore conclude that this first approach is of little practical interest.

4 Data Fusion through Intersection

4.1 Analysis of Results

Since the set of documents retrieved by the system is the same, the drop in performance in this first approach can only be caused by a worse ranking of the results because of the application of the distance-based model, and for this reason we decided to analyze the changes in the distribution of relevant and non-relevant documents in the K top retrieved documents. The results obtained in the case of using short queries and the triangle function (*tri*) are shown in Table 2. Changes in the type of query, short or long, or in the shape of the function, triangle or circle, has little effect on these results and the conclusions that can be inferred from them.

Each row contains the results obtained when comparing the K top documents retrieved by *lem* (set of results L), with those K top documents retrieved after their reranking using distances (set of results D). The columns show the results obtained for each of the parameters considered: average number of new relevant documents obtained through distances ($D \setminus L$), average number of relevant documents lost using distances ($L \setminus D$), average number of relevant documents preserved ($L \cap D$), overlap coefficient for relevant documents (R_{over}), precision of *lem* at K top documents ($Pr(L)$), precision at K top documents after reranking through distances ($Pr(D)$), precision for the documents common to both approaches in their K top documents ($Pr(L \cap D)$). The right-hand side of

Table 2. Document distribution (short queries - triangle function)

K	relevant docs. $D \setminus L$	$L \setminus D$	$L \cap D$	R_{over}	$Pr(L)$	$Pr(D)$	$Pr(L \cap D)$	non-relevant docs. $D \setminus L$	$L \setminus D$	$L \cap D$	N_{over}
5	1.60	1.62	1.44	0.47	**0.61**	**0.61**	***0.77***	1.54	1.52	0.42	0.22
10	2.73	2.89	2.81	0.50	**0.57**	**0.55**	***0.68***	3.14	2.98	1.32	0.30
15	3.37	3.77	4.22	0.54	**0.53**	**0.51**	***0.65***	5.13	4.73	2.28	0.32
20	3.92	4.44	5.59	0.57	**0.50**	**0.48**	***0.63***	7.24	6.72	3.25	0.32
30	4.66	5.62	7.99	0.61	**0.45**	**0.42**	***0.59***	11.84	10.88	5.51	0.33
50	5.95	9.16	20.69	0.73	**0.60**	**0.53**	***0.42***	45.04	41.83	28.32	0.39
100	5.10	7.84	31.78	0.83	**0.40**	**0.37**	***0.30***	88.13	85.39	74.99	0.46
200	1.99	2.82	45.72	0.95	**0.24**	**0.24**	***0.14***	164.28	163.45	288.01	0.64

the table shows their equivalents for the case of non-relevant documents: average number of non-relevant documents added, lost and preserved, together with their degree of overlap.

Several important facts can be observed in these figures. Firstly, that the number of relevant documents retrieved by both approaches in their K top documents is very similar —a little smaller for distances—, as can be inferred from the number of incoming and outgoing relevant documents, and from the precisions at the top K documents of both approaches. This confirms that the problem has its origin in a bad reranking of the results.

The second point we need to point out refers to the overlap coefficients of both relevant (R_{over}) and non-relevant (N_{over}) documents. These coefficients, defined by Lee in [12], show the degree of overlap among relevant and non-relevant documents in two retrieval results. For two runs run_1 and run_2, they are defined as follows:

$$R_{over} = \frac{2\ |Rel(run_1) \cap Rel(run_2)|}{|Rel(run_1)| + |Rel(run_2)|} \ . \tag{8}$$

$$N_{over} = \frac{2\ |Nonrel(run_1) \cap Nonrel(run_2)|}{|Nonrel(run_1)| + |Nonrel(run_2)|} \ . \tag{9}$$

where $Rel(X)$ and $Nonrel(X)$ represent, respectively, the set of relevant and non-relevant documents retrieved by the run X.

It can be seen in Table 2 that the overlap factor among relevant documents is much higher than among non-relevant documents. Therefore, it obeys the *unequal overlap property* [12], since both approaches return a similar set on relevant documents, but a different set on non-relevant documents. This is a good indicator of the effectiveness of fusion of both runs.

Finally, and also related with the previous point, the figures show that the precision for the documents common to both approaches in their K top documents ($Pr(L \cap D)$) is higher than the corresponding precisions for lemmas ($Pr(L)$) and distances ($Pr(D)$); that is, the probability of a document being relevant is higher when it is retrieved by both approaches. In other words, the more runs a document is retrieved by, the higher the rank that should be assigned to the document [17].

According to these observations, we decided to take a new approach for reranking, this time through data fusion, by combining the results obtained initially with the indexing of lemmas with the results obtained when they are reranked through distances. Next, we describe this approach.

4.2 Description of the Algorithm

Data fusion is a technique of combination of evidences that consists of combining the results retrieved by different representations of queries or documents, or by different retrieval techniques [7, 12, 4].

In our data fusion approach, we have opted for using a boolean criterion instead of combining scores based on similarities [7, 12] or ranks [12].

Once the value K is set, the documents are retrieved in the following order:

1. First, the documents contained in the intersection of the top K documents retrieved by both approaches: $L_K \cap D_K$. Our aim is to increase the precision of the top documents retrieved.
2. Next, the documents retrieved in the top K documents by only one of the approaches: $(L_K \cup D_K) \setminus (L_K \cap D_K)$. Our aim is to add to the top of the ranking those relevant documents retrieved only by the distance-based approach at its top, but without harming the ranking of the relevant documents retrieved by the indexing of lemmas.
3. Finally, the rest of documents retrieved using *lem*: $L \setminus (L_K \cup D_K)$.

where L is the set of documents retrieved by *lem*, L_K is the set of the top K documents retrieved by *lem*, and D_K is the set of the top K documents retrieved by applying distances.

With respect to the internal ranking of the results, we will take the ranking obtained with *lem* as reference, because of its better behavior. In this way, when a subset S of results is retrieved, they will be retrieved in the same relative order they had when they were retrieved by *lem*.[2]

5 Experimental Results with Data Fusion

After a previous phase of tuning of K, in which different values of K were tested[3], a value $K = 30$ was chosen as the best compromise, since although lower values of K showed peaks of precision in the top documents retrieved, their global behavior was worse.

Table 3 shows the results obtained with this new approach. Column *tri* shows the results obtained by means of the fusion through intersection of the set of documents initially retrieved with *lem* with the documents retrieved by applying reranking through distances using a triangle function. The results corresponding to the circle function are showed in *cir*.

[2] That is, if the original sequence in *lem* was *d2-d3-d1* and a subset {*d1,d3*} is going to be returned, the documents should be obtained in the same relative order as in the original results: *d3-d1*.

[3] $K \in \{5, 10, 15, 20, 30, 50, 75, 100, 200, 500\}$.

Table 3. Reranking through data fusion; K=30

	short queries				long queries			
	stm	*lem*	*tri*	*cir*	*stm*	*lem*	*tri*	*cir*
Documents	99k	99k	99k	99k	99k	99k	99k	99k
Relevant (5548 expected)	5086	5207	5207	5207	5208	5234	5234	5234
Non-interpolated precision	.5210	.5235	.5204	.5206	.5638	.5648	**.5654**	.5647
Document precision	.5502	.5814	**.5829**	**.5836**	.5925	.6038	**.6083**	**.6094**
R-precision	.4952	.4978	.4911	.4911	.5316	.5335	.5311	.5306
Precision at .00 recall	.8426	.8260	**.8424**	**.8428**	.9028	.8788	**.8871**	**.8901**
Precision at .10 recall	.7294	.7431	**.7520**	**.7522**	.7910	.7989	**.8052**	**.8075**
Precision at .20 recall	.6746	.6936	**.7043**	**.7059**	.7326	.7420	**.7501**	**.7496**
Precision at .30 recall	.6135	.6380	**.6434**	**.6447**	.6763	.6887	**.6975**	**.6983**
Precision at .40 recall	.5812	.5900	**.5967**	**.5965**	.6401	.6499	**.6577**	**.6595**
Precision at .50 recall	.5470	.5520	.5447	.5454	.5975	.6058	**.6092**	**.6141**
Precision at .60 recall	.5078	.5099	.4997	.4999	.5452	.5502	.5443	.5362
Precision at .70 recall	.4518	.4498	.4325	.4282	.4816	.4816	.4729	.4644
Precision at .80 recall	.3882	.3796	.3665	.3653	.4056	.4022	.3929	.3885
Precision at .90 recall	.3044	.2923	.2846	.2857	.3356	.3150	.3045	.3036
Precision at 1.0 recall	.1897	.1756	.1687	.1684	.2054	.1918	.1862	.1857
Precision at 5 docs	.6182	.6182	**.6303**	**.6343**	.6808	.6747	**.6929**	**.6949**
Precision at 10 docs	.5717	.5758	**.5929**	**.5970**	.6182	.6202	**.6525**	**.6495**
Precision at 15 docs	.5279	.5380	**.5522**	**.5542**	.5670	.5798	**.5993**	**.5980**
Precision at 20 docs	.4965	.5071	**.5217**	**.5207**	.5338	.5556	**.5672**	**.5646**
Precision at 30 docs	.4434	.4582	.4582	.4582	.4822	.5030	.5030	.5030
Precision at 100 docs	.2935	.3016	**.3040**	**.3044**	.3119	.3171	**.3182**	**.3193**
Precision at 200 docs	.1937	.2002	**.2006**	**.2008**	.2053	.2060	**.2064**	**.2064**
Precision at 500 docs	.0945	.0981	**.0982**	**.0982**	.0981	.0985	**.0987**	**.0987**
Precision at 1000 docs	.0514	.0526	.0526	.0526	.0526	.0529	.0529	.0529

The improvements attained with this new approach —in boldface— are general, particularly in the case of the precision at N documents retrieved. Moreover, there are no penalizations for non-interpolated precision and R-precision.

6 Conclusions and Future Work

In this article we have proposed the use of a distance-based retrieval model, also called locality-based, which allows us to face the problem of syntactic linguistic variation in text conflation employing a pseudo-syntactic approach.

Two approaches were proposed for this purpose, both based on reranking the results obtained by indexing content word lemmas. The first approach, where the ranking obtained by means of the application of the locality-based model is the final ranking to be retrieved, did not get, in general, good results. After analyzing the behavior of the system, a new approach was taken, this time based on data fusion, which employs the intersection of the sets of documents retrieved by both approaches as reference for the reranking. This second approach was fruitful, since it obtained consistent improvements in the ranking at all levels, without harming other aspects.

With respect to future work, several aspects should be studied. Firstly, we intend to extend our experiments to other retrieval models apart from the vector model, in order to test its generality. Secondly, we aim to improve the system by managing not only syntactic variants but also morphosyntactic variants [9].

Two new applications of this locality-based approach are also being considered. Firstly, in Query Answering, where it will in all probability prove most useful, since this distance-based model allows us to identify the relevant locations of a document, which probably contain the answer, with respect to the query. Once the relevant locations are identified, the answer would be extracted through further in-depth linguistic processing. Secondly, its possible application in query expansion through *local clustering based on distances* [2] is also being studied.

Acknowledgements

The research reported in this article has been partially supported by Ministerio de Ciencia y Tecnología (HF2002-81), FPU grants of Secretaría de Estado de Educación y Universidades (AP2001-2545), Xunta de Galicia (PGIDIT02PXIB30501PR, PGIDIT02SIN01E and PGIDIT03SIN30501PR) and Universidade da Coruña.

References

1. M. A. Alonso, J. Vilares, and V. M. Darriba. On the usefulness of extracting syntactic dependencies for text indexing. In M. O'Neill, F. F. E. Sutcliffe, C. Ryan, and M. Eaton, editors, *Artificial Intelligence and Cognitive Science*, volume 2464 of *Lecture Notes in Artificial Intelligence*, pages 3–11. Springer-Verlag, Berlin-Heidelberg-New York, 2002.
2. R. Attar and A.S. Fraenkel. Local feedback in full-text retrieval systems. *Journal of the ACM*, 24(3):397–417, July 1977.
3. C. Buckley. Implementation of the SMART information retrieval system. Technical report, Department of Computer Science, Cornell University, 1985. Sources available in `ftp://ftp.cs.cornell.edu/pub/smart`.
4. W. B. Croft. Combining approaches to information retrieval. In W. B. Croft, editor, *Advances in Information Retrieval. Recent Research from the Center for Intelligent Information Retrieval*, volume 7 of *The Kluwer International Series on Information Retrieval*, chapter 1, pages 1–36. Kluwer Academic Publishers, Boston/Dordrecht/London, 2000.
5. O. de Kretser and A. Moffat. Effective document presentation with a locality-based similarity heuristic. In *Proc. of the 22nd annual international ACM SIGIR conference on Research and Development in Information Retrieval*, pages 113–120, Berkeley, California, USA, 1999. ACM Press, New York.
6. O. de Kretser and A. Moffat. Locality-based information retrieval. In J. F. Roddick, editor, *Proc. of 10th Australasian Database Conference (ADC '99), 18-21 January, Auckland, New Zealand*, volume 21 of *Australian Computer Science Communications*, pages 177–188, Singapore, 1999. Springer-Verlag.
7. E. Fox and J. Shaw. Combination of multiple searches. In D. K. Harman, editor, *NIST Special Publication 500-215: The Second Text REtrieval Conference (TREC-2)*, pages 243–252, Gaithersburg, MD, USA, 1994. Department of Commerce, National Institute of Standards and Technology.

8. D. A. Hull, G. Grefenstette, B. M. Schulze, E. Gaussier, H. Schütze, and J. O. Pedersen. Xerox TREC-5 site report: Routing, filtering, NLP, and Spanish tracks. In E. M. Voorhees and D. K. Harman, editors, *NIST Special Publication 500-238: The Fifth Text REtrieval Conference (TREC-5)*, pages 167–180, Gaithersburg, MD, USA, 1997. Department of Commerce, National Institute of Standards and Technology.
9. C. Jacquemin and E. Tzoukermann. NLP for term variant extraction: synergy between morphology, lexicon and syntax. In *Natural Language Information Retrieval*, volume 7 of *Text, Speech and Language Technology*, pages 25–74. Kluwer Academic Publishers, Dordrecht/Boston/London, 1999.
10. M. Kaszkiel and J. Zobel. Effective ranking with arbitrary passages. *Journal of the American Society of Information Science*, 52(4):344–364, 2001.
11. C. H. Koster. Head/modifier frames for information retrieval. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 2945 of *Lecture Notes in Computer Science*, pages 420–432. Springer-Verlag, Berlin-Heidelberg-New York, 2004.
12. J. Lee. Analyses of multiple evidence combination. In *Proc. of SIGIR '97, July 27-31, Philadelphia, PA, USA*, pages 267–276. ACM Press, 1997.
13. M. Mitra, C. Buckley, A. Singhal, and C. Cardie. An analysis of statistical and syntactic phrases. In L. Devroye and C. Chrisment, editors, *Proc. of Computer-Aided Information Searching on the Internet (RIAO'97)*, pages 200–214, Montreal, Canada, 1997.
14. J. Perez-Carballo and T. Strzalkowski. Natural language information retrieval: progress report. *Information Processing and Management*, 36(1):155–178, 2000.
15. C. Peters, editor. *Results of the CLEF 2002 Cross-Language System Evaluation Campaign, Working Notes for the CLEF 2002 Workshop*, Rome, Italy, Sept. 2002. CLEF official site: `http://www.clef-campaign.org`.
16. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The SMART Retrieval System - Experiments in Automatic Document Processing*, pages 313–323. Prentice-Hall, Englewood Cliffs, NJ, 1971.
17. T. Saracevic and P. Kantor. A study of information seeking and retrieving. III. Searchers, searches, overlap. *Journal of the American Society for Information Science*, 39(3):197–216, 1988.
18. J. Vilares and M. A. Alonso. A grammatical approach to the extraction of index terms. In G. Angelova, K. Bontcheva, R. Mitkov, N. Nicolov, and N. Nikolov, editors, *International Conference on Recent Advances in Natural Language Processing, Proceedings*, pages 500–504, Borovets, Bulgaria, Sept. 2003.
19. J. Vilares, M.A. Alonso, and F.J. Ribadas. COLE experiments at CLEF 2003 Spanish monolingual track. To be published in *Lecture Notes in Computer Science*. Springer-Verlag, Berlin-Heidelberg-New York, 2004.
20. J. Vilares, M.A. Alonso, F.J. Ribadas, and M. Vilares. COLE experiments at CLEF 2002 Spanish monolingual track. In C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors, *Advances in Cross-Language Information Retrieval*, volume 2785 of *Lecture Notes in Computer Science*, pages 265–278. Springer-Verlag, Berlin-Heidelberg-New York, 2003.