Syntactic and pseudo-syntactic approaches for text retrieval

Jesús Vilares, Carlos Gómez-Rodríguez and Miguel A. Alonso
Departamento de Computación, Universidade da Coruña, Campus de Elviña, 15071 - A Coruña, SPAIN
{jvilares, cgomezr, alonso}@udc.es - www.grupocole.org

We study how the use of syntactic information can improve the performance of Information Retrieval systems based on single-word terms. We consider two different approaches. The first one identifies the syntactic structure of the text by means of a shallow parser in order to extract the head-modifier pairs of the most relevant syntactic dependencies, which are used as complex index terms. The second approach uses pseudo-syntactic information based on the distance between terms, taken as work hypothesis that there exist a fuzzy relation between the proximity of two terms and the fact that both terms are linked by a syntactic relation. Both approaches have been tested on the CLEF collections.

Keywords: Information Retrieval, syntactic linguistic variation, shallow parsing, locality-based model.

1 INTRODUCTION

Natural Language Processing (NLP) has attracted the attention of the Information Retrieval (IR) community since one of the major limitations of IR systems is linguistic variation, that is, the different ways in which the same concept can be expressed [3]. In this context, syntactic processing has been applied for dealing with the syntactic variation present in natural language texts, 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 –wide-coverage grammars, treebanks, etc.– , as in the case of Spanish, the application of these techniques is a real challenge.

Taking into account these limitations, we propose in this paper two alternative approaches to the problem. Our first approach tries to obtain the syntactic structure of the text through shallow parsing in order to process its syntactic content. As a second alternative, we propose the use of pseudo-syntactic information based on the distance between terms, considering as working hypothesis that there exist a fuzzy relation between the proximity of two terms and the fact that both terms are linked by a syntactic relation (as a consequence of its fuzzyness, counterexamples exist, e.g., long distance dependencies).

2 A SYNTACTIC APPROACH TO TEXT RETRIEVAL

When processing the syntactic content of a text, the first step consists in obtaining its syntactic structure. Nevertheless, full parsing of the text is non-viable here because of its high computational cost –which makes its application on a large scale impractical– and its lack of robustness –which greatly reduces their coverage, particularly in the case of Romance languages, due to the lack of freely available resources for most of them. In this context, the employment of *shallow parsing* techniques [1] allows us to reduce computational complexity and increase robustness.

The shallow parser employed by our system consists of five layers, the input of the parser being the output of a tagger-lemmatizer [6]. Each of the rules involved in the different stages of the parsing process has been implemented through a finite-state transducer, compounding, in this way, a finite-state cascade-based parser which maintains a linear complexity. The first layer, *layer 0*, preprocesses quantity and verbal expressions. Next, *layer 1* identifies adverbial phrases and non-periphrastic verbal groups, *layer 2* whilst deals with adjectival phrases and periphrastic verbal groups. *Layer 3* manages noun phrases. Finally, *layer 4* processes prepositional phrases. These layers and the grammar rules employed by the parser are explained in detail in [12].

Our goal is to obtain those pairs of words related through the most significative syntactic dependencies, those of the type *noun-modifier*, *subject-verb*, *verb-object*, etc. Once the dependencies have been extracted, they are conflated into *complex index terms* [7] in order to complement simple terms, since their degree of specificity is greater than those for their individual component terms. In our case, we have used a conflation technique based on the use of morphological relations in order to improve the management of syntactic variation [2] by covering both the syntactic and morphosyntactic variants of a term [7] –e.g., *una caída de las ventas* (a drop in the sales) vs. *una caída de ventas* (a drop in sales) vs. *las ventas han caído* (sales have dropped).

Table 1. Experiments using the CLEF corpus

Method	*stm*	*tsd*	*dsd*	stm_f	tsd_f	dsd_f	*cir*
# ret. docs.	46k	46k	46k	46k	46k	46k	46k
# rel. ret. docs.	2719	2728	2758	2606	2811	2780	2767
Non-int. Pr.	0.4720	0.4965	0.5286	0.5032	0.5434	0.5382	0.5327
R-Pr.	0.4599	0.4895	0.5119	0.4796	0.5210	0.5097	0.5126
Pr. at 5	0.6391	0.6913	0.7000	0.6391	0.6913	0.7043	0.6739
Pr. at 10	0.5935	0.6500	0.6717	0.6087	0.6739	0.6804	0.6761
Pr. at 15	0.5551	0.6029	0.6203	0.5609	0.6246	0.6362	0.6188
Pr. at 20	0.5174	0.5620	0.5935	0.5478	0.5946	0.6000	0.5826
Pr. at 30	0.4710	0.5036	0.5348	0.5000	0.5543	0.5319	0.5225
Pr. at 100	0.3157	0.3348	0.3474	0.3274	0.3550	0.3372	0.3502
Pr. at 200	0.2186	0.2263	0.2336	0.2157	0.2357	0.2258	0.2349
Pr. at 500	0.1097	0.1103	0.1117	0.1045	0.1135	0.1114	0.1122
Pr. at 1000	0.0591	0.0593	0.0600	0.0567	0.0611	0.0604	0.0602

2.1 Experimental Results

Our approaches have been integrated in the well-known vector-based engine SMART[1], using an `atn-ntc` weighting scheme. The evaluation was made using the Spanish corpus employed in CLEF 2001/02 editions[2], formed by 215,738 news reports (509 MB), and the 46 odd-numbered topics of those editions (41 to 140) with more than 5 relevant documents. These topics are formed by three fields: a brief *title* statement, a one-sentence *description*, and a more complex *narrative* specifying the relevance assessment criteria. Only *title* and *description* statements have been employed.

We have compared the behavior of three different approaches:

- *Stemming (*stm*).* Our baseline, it employes the Snowball Spanish stemmer[3], based on Porter's algorithm.
- *Syntactic dependency pairs obtained from the topic (*tsd*).* A NLP-based approach in which documents are indexed taking into account the lemmas of the content-words (nouns, adjectives and verbs) and the complex terms derived from the syntactic dependencies. The query submitted to the system is formed by the index terms obtained from the topic through the same process of lemmatization and shallow parsing applied to documents.
- *Syntactic dependency pairs obtained from top documents (*dsd*).* A NLP-based approach. The indexing process is the same of *tsd*, but the querying process is performed in two stages [14]:
  1. The lemmatized query is submitted to the system.
  2. The n top documents retrieved by this initial query are employed to select the most informative dependencies, which are used to expand the lemmatized query, but with no re-weighting. These dependencies are selected automatically from the 50 best terms (both lemmas and dependencies) of the top 10 documents using Rocchio's approach to feedback [10].

  The expanded query is then submitted to the system in order to obtain the final set of documents retrieved.

The first group of columns of Table 1 shows the results for this first set of experiments. Each row of the table contains one of the parameters employed to measure the performance: number of documents retrieved by the system, number of relevant documents retrieved (3007 expected), non-interpolated precision, R-precision, and precision at N documents retrieved.

The results obtained with our NLP-based approaches (*tsd* and *dsd*) in this first set of experiments show a clear and consistent improvement at all levels with respect to classical stemming (*stm*). Moreover, document dependencies (*dsd*) show a improved behavior not only with respect to stemming (*stm*), but also with respect to topic dependencies (*tsd*). We can conclude from these results that the pairs chosen automatically by the system seem to be more accurate than those obtained directly from the topic.

The second group of columns of Table 1 (stm_f, tsd_f and dsd_f) correspond to a second set of experiments performed in order to compare the behavior of syntactic approaches with respect to stemming when using pseudo-relevance feedback (blind-query expansion). For these tests we have adopted Rocchio's

[1] `ftp://ftp.cs.cornell.edu/pub/smart`
[2] `http://www.clef-campaign.org`
[3] `http://snowball.tartarus.org`

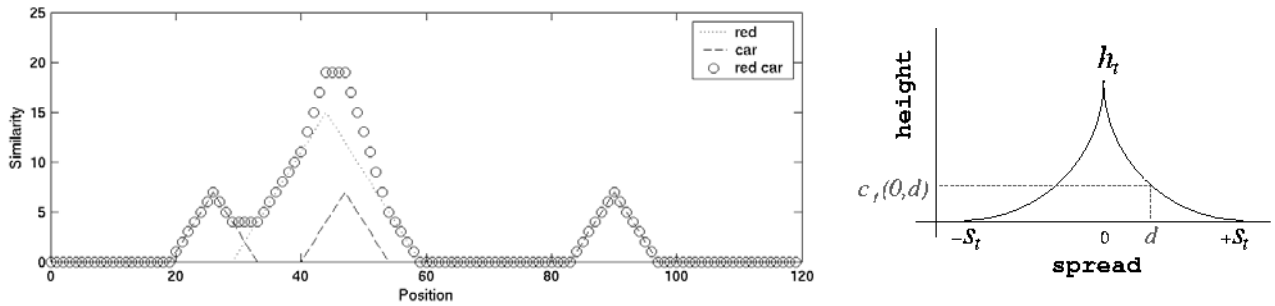

Fig. 1. *(Left)* Computing the similarity measure in a locality-based model for the query *"red car"*: positions where query terms occur and their regions of influence, and the resultant similarity curve. *(Right)* Similarity contribution function c_t for the circle shape

approach [10].[4] As we can see, our NLP-based approaches show again a consistent improvement with respect to stemming. Nevertheless, the differences between the employment of topic or query dependencies are now lesser than before, since document dependencies (dsd_f) outperform topic dependencies –tsd_f– only for the precision of the first documents retrieved. For other measures, topic dependencies (tsd_f) obtain better results.

3 A PSEUDO-SYNTACTIC APPROACH BASED ON THE DISTANCE BETWEEN TERMS

The *locality-based model* [8] 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 text 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 is shown in the left part of Fig. 1.

The contribution to the similarity graph of a given query term is determined by a *similarity contribution function* c_t –see right part of Fig. 1– defined according to 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 (i.e., the scope of its influence), and 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 [8], but we only show here that one with which we obtained better results in Spanish, the circle function (*cir*), defined by $c_t(x,l) = h_t\sqrt{1-(d/s_t)^2}$, with $c_t(x,l) = 0$ when $|x-l| > s_t$ –see right part of Fig. 1.

The height h_t of a query term t is defined as an inverse function of its frequency in the collection $h_t = f_{q,t}\log_e(N/f_t)$, 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 $s_t = {}^{n}\!/_{N}\,{}^{N}\!/_{f_t} = {}^{n}\!/_{f_t}$, 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

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

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

[4] Expansion through the 10 best terms of the 5 top documents retrieved with weights α=0.80, β=0.10 and γ=0.

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. 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.

3.1 Experimental Results

In our approach, the locality-based model is used to postprocess the documents retrieved by a conventional document-based retrieval system.[5] This initial set of documents is obtained through a base IR system which employs content-word lemmas as index terms. The list of documents returned is then processed using the locality-based model.

Instead of the original iterative algorithm [8], 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:

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

Initial results obtained by employing this score as the final score to be retrieved to the user [13] showed a general drop in performance, except for low recall levels, where results were similar and sometimes even better. We decided to analyze the changes in the distribution of relevant and non-relevant documents in the *K* top retrieved documents of both the original and the post-processed document ranking. For this purpose, we studied the Lee's overlap coefficients [9] of both relevant (R_{over}) and non-relevant (N_{over}) documents. For two runs run_1 and run_2, they are defined as follows:

$$R_{over} = \frac{2\left|Rel(run_1) \cap Rel(run_2)\right|}{\left|Rel(run_1)\right| + \left|Rel(run_2)\right|}. \qquad N_{over} = \frac{2\left|Nonrel(run_1) \cap Nonrel(run_2)\right|}{\left|Nonrel(run_1)\right| + \left|Nonrel(run_2)\right|}.$$

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

We observed that the overlap factor among relevant documents was much higher than among non-relevant documents. Therefore, it obeyed the *unequal overlap property* [9], since both runs returned a similar set of relevant documents, but a different set of non-relevant documents. This is a good indicator of the effectiveness of fusion of both runs. We also observed that the precision for the documents common to both runs in their *K* top documents was higher than the corresponding precisions for lemmas and distances, that is, the probability of a document being relevant was higher when it was 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 [11].

According to these observations, we decided to take a new approach for reranking, this time through data fusion [5][9], by combining the results obtained initially with the indexing of lemmas with the results obtained when they are reranked through distances. So, once a value *K* is set[6], the documents are retrieved in the following order. First, the documents contained in the intersection of the top *K* documents retrieved by each run: our aim is to increase the precision of the top documents retrieved. Next, the documents retrieved in the top *K* documents by only one of the runs: 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 those retrieved by the indexing of lemmas. Finally, the rest of documents retrieved using lemmas.

Last column of Table 1 (*cir*) shows the results obtained with this new approach. As we can observe, both syntactic and pseudo-syntactic methods beat stemming at all levels. Nevertheless, syntactic methods show slightly better results than those obtained with our pseudo-syntactic approach based on distances, but at the expense of needing of a grammar and a parser, while our distance-based approach needs no grammar or parser at all.

[5] Similar two-round experiments have not been effective when document-based IR models have been applied in both rounds.

[6] A value $K=30$ was chosen after a tuning phase trying $K \in \{5, 10, 15, 20, 30, 50, 75, 100, 200, 500\}$.

4 CONCLUSIONS

In this article we have studied the employment of several techniques in text retrieval for dealing with syntactic variation. Two different approaches have been tested. Firstly, the syntactic approach employs a shallow parser to extract the syntactic dependencies present in queries and documents in order to obtain the corresponding head-modifier pairs, which are used as multi-word terms to build the index of the text retrieval system. Secondly, the pseudo-syntactic approach uses a distance-based retrieval model, also called locality-based, to take into account that close terms are often related, at some degree, by a syntactic relation, without needing a grammar or parser.

Both techniques have been originally designed for Spanish, but their general architecture can be easily adapted to other Romance languages. Both approaches have been tested on the Spanish CLEF collection and show a similar performance, improving the results obtained by indexing techniques based on single-word terms. Shallow-parsing-based approaches show slightly better results, but at the expense of needing a grammar and a parser. On the other hand, our distance-based does not need them.

As future work, we are investigating the possibility of incorporating more elaborated forms of robust parsing and a fuzzy notion of synonymy [4].

ACKNOWLEDGMENT

The research reported in this article has been partially supported by Ministerio de Educación y Ciencia and FEDER (Grant TIN2004-07246-C03-02), Xunta de Galicia (Grants PGIDIT05PXIC30501PN, PGIDIT05PXIC10501PN, PGIDIT05SIN044E), and Secretaría de Estado de Universidades e Investigación (FPU grants).

REFERENCES

[1] Steven Abney. Partial parsing via finite-state cascades. *Natural Language Engineering*, 2(4):337-344, 1997.

[2] Miguel A. Alonso, Jesús Vilares, and Víctor M. Darriba. On the usefulness of extracting syntactic dependencies for text indexing. In vol. 2464 of *Lecture Notes in Artificial Intelligence*, pp. 3-11. Springer-Verlag, Berlin, 2002.

[3] Avi Arampatzis, Th. P. van der Weide, P. van Bommel, and C.H.A. Koster. Linguistically-motivated Information Retrieval. In *Encyclopedia of Library and Information Science*, vol. 69, pp. 201-222. Marcel Dekker, Inc, New York-Basel, 2000.

[4] Santiago Fernández-Lanza, Jorge Graña, and Alejandro Sobrino. Introducing FDSA (Fuzzy Dictionary of Synonyms and Antonyms): Applications on Information Retrieval and Stand-Alone Use. *Mathware & Soft Computing*, 10(2-3):57-70, 2003.

[5] Edward A. Fox and Joseph A. Shaw. Combination of multiple searches. In *Proc. of 2nd Text REtrieval Conference (TREC-2), August 31-September 2, 1993, Gaithersburg, MD, USA*, pp. 243-252. National Institute of Standards and Technology Special Publication 500-215, 1994.

[6] Jorge Graña. *Técnicas de Análisis Sintáctico Robusto para la Etiquetación del Lenguaje Natural*. PhD thesis, Departamento de Computación, Universidade da Coruña, A Coruña, Spain, December 2000.

[7] Christian Jacquemin and Evelyne Tzoukermann. NLP for term variant extraction: synergy between morphology, lexicon and syntax. In Tomek Strzalkowski, editor, *Natural Language Information Retrieval*, vol. 7 of *Text, Speech and Language Technology*, pp. 25-74. Kluwer Academic Publishers, Dordrecht/Boston/Londres, 1999.

[8] O. de Kretser and A. Moffat. Locality-based information retrieval. In *Proc. of 10th Australasian Database Conference (ADC '99), 18-21 January, Auckland, New Zealand*, pp. 177-188, 1999.

[9] Joon Ho Lee. Analyses of multiple evidence combination. In *Proc. of SIGIR'97, July 27-31, Philadelphia, PA, USA*, pp. 267-276. ACM Press, 1997.

[10] J.J. Rocchio. *The SMART Retrieval System - Experiments in Automatic Document Processing*, chapter Relevance feedback in information retrieval, pp. 313-323. Prentice-Hall, Englewood Cliffs, NJ, 1971.

[11] 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.

[12] Jesús Vilares and Miguel A. Alonso. A grammatical approach to the extraction of index terms. In *Proc. of International Conference on Recent Advances in Natural Language Processing (RANLP 2003), 10-12 September, Borovest, Bulgaria*, pp. 500-504, 2003.

[13] Jesús Vilares and Miguel A. Alonso. Dealing with syntactic variation through a locality-based approach. In vol. 3246 of *Lecture Notes in Computer Science*, pp. 255-266. Springer-Verlag, Berlin, 2004.

[14] Jesús Vilares, Miguel A. Alonso, and Manuel Vilares. Morphological and syntactic processing for text retrieval. In vol. 3180 of *Lecture Notes in Computer Science*, pp. 371-380. Springer-Verlag, Berlin, 2004.