QUERY EXPANSION USING WORDNET WITH A LOGICAL MODEL OF INFORMATION RETRIEVAL

David Parapar, Álvaro Barreiro
AILab, Department of Computer Science,
University of A Coruña, Spain
dparapar@udc.es, barreiro@udc.es

David E. Losada
Intelligent Systems Group, Department of Electronics and Computer Science,
University of Santiago de Compostela, Spain
dlosada@usc.es

ABSTRACT

This paper describes the experimentation conducted to test the effectiveness of query expansion within the logical model PLBR. We ran different experiments generating queries as logical formulas with different connectives, and using different types of linguistic information extracted from WordNet. Results show that lexical expansion is not able to improve retrieval performance. Nevertheless, the experiments allow us to conclude that query expansion can benefit from a logical model which allows structured queries.

KEYWORDS

Information retrieval, query expansion, WordNet, logical models of IR.

1. INTRODUCTION

The primary objective of this work is to determine whether or not query expansion yields any benefit in the context of a logical mode of information retrieval. The PLBR model, based on Propositional Logic and Belief Revision is a good starting point because it has been evaluated in the past but no previous expansion experiments were run. To simplify the test, the experiments only used the linguistic information recorded in WordNet as the source for expansion terms.

The expansion experiments were tested against a subset of the TREC collection[13]. Each initial query in every experiment was generated automatically from the TREC topics. Query words are then selected and expanded using the lexical relations included in WordNet. The selection of the correct meaning of each word was done manually, because our main interest is to evaluate logical query expansion and not an automatic disambiguation of senses. Thus the effect of expansion can be analyzed without regard to the quality of the disambiguation, which is assumed to be optimum. In addition, we select the words only from the title of the topic. This simulates a common user query with few and generic words.

The pioneer work in query expansion using WordNet was made by Voorhees[15]. The results of this work were not good, especially when initial queries are long. In the case of initial short queries, query effectiveness was improved but it was not better than the effectiveness achieved with complete long queries without expansion. Smeaton et al.[12] used the concept of specificity of words, and expanded specific terms with the parents and grandparents in the WordNet hierarchy and the abstract terms with the children and grandchildren. Furthermore, every word is expanded with its synonyms. The results in terms of precision were disappointing.

In the work of Mandala et al.[7] the relations stored in WordNet are combined with similarity measures based on syntactic dependencies and co-occurrence information. This combination improves the effectiveness of retrieval. The work of Qiu and Frei[11] used an automatically constructed thesaurus and the results were good but the expansion was tested against small document collections. Other successful works

used thesaurus adapted with relevance information (Nie[9]) or were tested against collections in specific domains. We will use only linguistic information and we will test expansion with a large subset of the TREC collection.

In these works the retrieval model used is the *Vector Space Model* (VSM). In this model the queries are represented as vectors. The expansion consists in the simply addition of terms to the vector. Nie and Jin[10] found a problem in this direct addition of terms in VSM. Let us consider a query $<a,b>$. If there are three expansion terms a_1, a_2, a_3 related with a and one expansion term b_1 related with b, the expanded query $<a, b, a_1, a_2, a_3, b_1>$ will give more importance to the concept associated with a than to the concept associated with b. The retrieved documents will probably satisfy aspect a more than aspect b. Nie and Jin proposed a vector representation that integrate the logical OR relation. For the evaluation of the logical OR in the framework of the VSM, two alternatives were considered. The first alternative is the transformation of the expanded query into a logical combination (with OR) of vectors. The second alternative is the direct evaluation of each dimension in the expanded query. For a dimension where a logical OR exists a single similarity value is calculated using fuzzy logic metrics. They only implemented the second alternative.

The PLBR model allows us to implement the first alternative including the OR operator in a natural way. In addition, the expansion in the PLBR framework could also be made with the connective AND, and allows the inclusion of negative terms. The expansion using these alternatives yields structured queries that can be compared with their unstructured counterparts. In addition, the negative expansion allows to include information about documents that we do not want to retrieve.

In the next section we will examine the PLBR model explaining the representation of documents and queries and the model for matching. Next we briefly recall what WordNet is. In section 4 we will present the application developed for formatting queries and designing experiments. Section 5 describes the experiments and their results. The conclusions are presented in the last section.

2. THE PLBR MODEL

This section describes briefly the basic foundations of the PLBR model, further details can be found elsewhere[2,3].

In this model documents and queries are represented as Propositional Logic Formula which are constructed from an alphabet of terms using the logical connectives $\wedge$ (conjunction), $\vee$ (disjunction) and $\neg$ (negation). Initially these formulas could have any form but for efficiency reasons they must be translated into disjunctive normal form (DNF). Given a document and a query represented by the propositional formulas d and q respectively, the application of the notion of logical consequence to decide relevance, i.e. $d \models q$, is too strict[14]. The entailment $d \models q$ simply tests whether or not each logical representation that makes d true makes also q true (i.e. each *model* of d is also a model of q). This is not in accordance with what we expect from an IR measure of relevance. In the next example we have two documents represented as $d_1 = a \wedge b \wedge \neg c \wedge d$ and $d_2 = \neg a \wedge \neg b \wedge \neg c \wedge d$ and a query represented as $q = a \wedge b \wedge c$. Both documents fail to fulfill the entailment, i.e. both $d_1 \models q$ and $d_2 \models q$ do not hold. This is because there exist models of d_1 and d_2 that map the query into false. As a consequence, the application of the logical entailment to decide relevance would assign the same status to both documents with respect to the query q. This is not appropriate for IR purposes because d_1 is likely more relevant than d_2 (d_1 fulfills partially the query).

In [4] a method to get a non-binary measure of the entailment $d \models q$ was proposed. To define a non-binary measure of relevance the distance from each model of d to the set of models of q is measured. In the field of Belief Revision (BR) measures of distance between logical interpretations are formally defined. The basic BR problem can be defined as follows. Let T be a logical theory and A a new formula to be included in the theory. BR methods define a way to include the new information in the theory. If there is no contradiction between T and A, the solution is trivial because the new theory, $T \circ A$ ($\circ$ stands for a revision operator), is just $T \wedge A$. However, if contradiction arises some old knowledge has to be removed in order to get a consistent new theory. Basically, a measure of closeness to the set of models of the theory T is defined and the models of A which are closest to the models of T are chosen to be the models of the new theory. As a consequence, BR model-based approaches are suitable for measuring distances from documents to queries when both are represented as logical formulas. Next paragraph sketches the details of this formulation.

In [4] there was found an interesting connection between Dalal's BR operator[1], $\circ_D$, and IR matching functions. Let us regard a query *q* as a logical theory and a document *d* as a new information. In the revision process q $\circ_D$ d, a measure of distance from a given document interpretation to the set of models of the query is defined. An important circumstance is that the semantics of this measure is appropriate for IR. Given a model of the document, the measure represents the number of propositional letters (i.e. index terms) that should be changed in that model in order to satisfy the query. For instance, let us consider an alphabet composed of three terms (*neural, science, network)* and a complete document *d* (i.e. a document having a single model) represented as $neural \wedge science \wedge \neg network$ and a query *q* represented as $neural \wedge network$. The distance from the document to the query would be equal to one because we would change the truth value of one propositional letter in the document (*network*) in order to satisfy the query. For that hypothetical changed document *d'*, $d' \models q$ would hold.

In the general case a document representation may be partial, hence, there might be several interpretations in which the document is satisfied (i.e. several document models). In order to get a non-binary measure of the entailment $d \models q$ we can compute the distance from each model of the document to the set of models of the query and, finally, calculate the average over document's models. This average over document's models is translated into a similarity measure, *BRsim*, in the interval [0,1].

Because *BRsim* is model-based, a direct computation would require exponential time (the number of logical interpretations grows exponentially with the size of the alphabet). In [5,6] efficient procedures to approximate the computation of *BRsim* were proposed. A restriction in the syntactical form of the logical formulas involved allows to design polynomial-time algorithms to compute similarity. Specifically, the propositional formulas representing documents and queries have to be in disjunctive normal form (DNF). A DNF formula has the form: $c_1 \vee c_2 \vee \ldots$ where each c_j is a conjunction of literals (also called *conjunctive clause*): $l_1 \wedge l_2 \wedge \ldots$. A literal is a propositional letter or its negation. As a result, a document *d* and a query *q* can be efficiently matched as long as *d* and *q* are in DNF. This restriction is acceptable because the expressiveness of generic propositional formulas and DNF formulas is the same. Indexing procedures have to represent documents as DNF formulas. From the user perspective, the use of DNF formulas does not introduce additional penalties. A translation from a natural language information need into a DNF query can be done automatically or, alternatively, users can be asked to write propositional formulas and a translation into DNF is automatically done.

Let us imagine a document *d* represented by a DNF formula $dc_1 \vee dc_2 \vee \ldots$ and a query *q* represented by a DNF formula $qc_1 \vee qc_2 \vee \ldots$, where each dc_i (qc_i) is a conjunctive clause. The distance from the document to the query is measured as the average distance from document clauses to the set of query clauses. The distance from an individual document clause dc_j to the set of query clauses is measured as the minimum distance from dc_j to the query clauses. Intuitively, different query clauses represent different requirements in the information need and the distance from dc_j to the query is measured as the distance to the requirement(s) that dc_j best fulfills. The clause-to-clause distance depends on (1) the number of literals appearing as positive literals within one clause and as negative literals within the other clause and (2) the number of the literals in the query clause whose propositional letter is not mentioned by the document clause. The clause-to-clause distance helps to determine to determine how good is the document clause for satisfying the query clause. In this respect, a contradicting literal, case (1), produces an increment of the distance greater than a query literal not mentioned by the document, case (2). This is because we do not know whether or not the document clause actually deals with that term (documents representations are partial: information about presence/absence is not available for all the terms in the alphabet).

3. WORDNET

WordNet[8] is a lexical system manually constructed by a group of people leaded by George Miller at the Cognitive Science Laboratory at Princeton University. WordNet is organized in sets of synonyms (*synsets*) with the words with the same meaning. These synsets have different relations between them. The relation of hypernymy/hyponymy (is-a relation) is the principal relation and creates a hierarchic structure. There are also relations of meronymy/holonymy (part-of relation). In addition, WordNet is divided in four taxonomies by the type of word: nouns, verbs, adjectives and adverbs. We only used the taxonomy of nouns because nouns

are the most content-bearing words. Expansion terms will be selected from the correct synsets for each noun in the query.

4. QUERY FORMULATION

Queries are generated starting from the TREC topics, selecting the expansion terms and fetching lexical information from WordNet. We developed a web application that let us to test different options of expansion.

In the definition of an experiment we can choose among different options as we can see in figure 1. First we select the topic or topic range over which we are going to generate the queries, the stemming algorithm, the taxonomies of WordNet and the level of expansion (synonyms and/or first level hyponyms).

The initial noun-expanded query can be formed from different fields on the TREC topic (title, description and narrative) and the specific terms to be included in the initial query can be selected either manual or automatically. In the case of automatic selection a stop list can be loaded from removing common words. The terms selected for the initial queries could be connected before expansion with $\vee$ or $\wedge$. In the following, in the context of the interface of the application for query formulation, we write OR/AND instead of $\vee/\wedge$.

The last options in the figure are about expansion strategies. We can expand a query with the correct synset of each word adding its terms with OR or AND. For example, for the original query $t_1 \wedge t_2$ if we have only a term t_1' related with t_1 the expanded query will be $(t_1 \vee t_1') \wedge t_2$ or $(t_1 \wedge t_1') \wedge t_2$. We can also expand with terms selected from the incorrect synsets and incorporate then in the expanded query as negated terms. In this case we can select the incorrect synsets manually or simply choose the correct synset and consider the remaining synsets as incorrect.

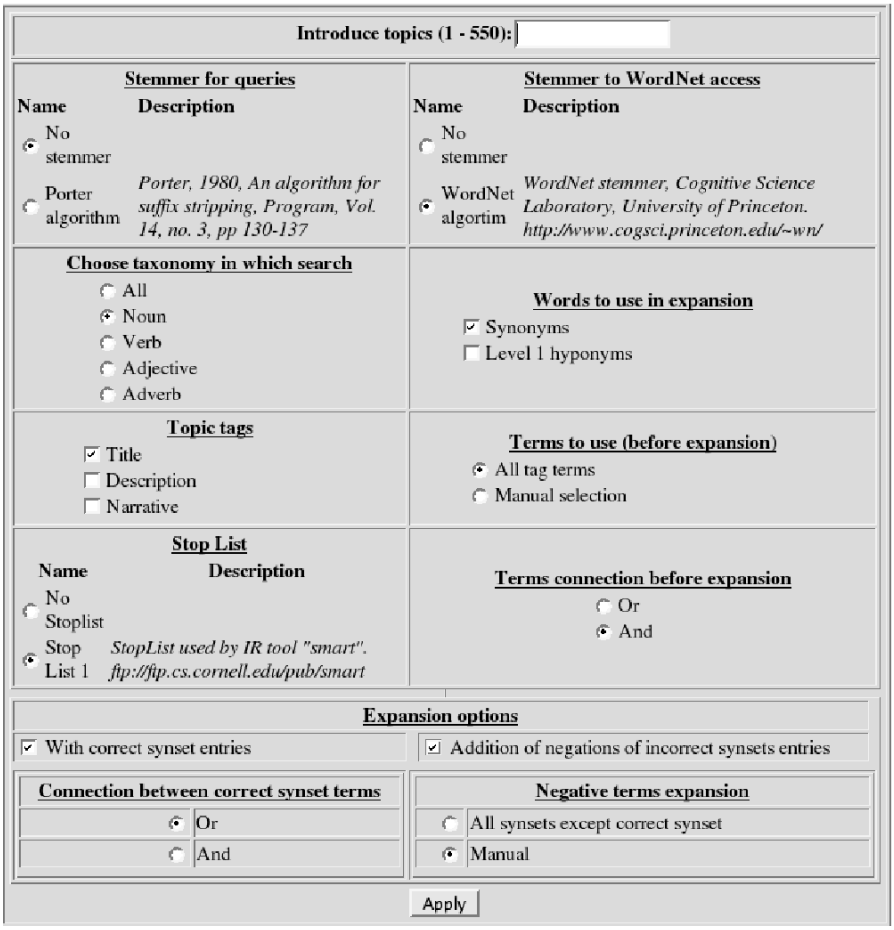

Figure 1. Interface of the application which allows to define expansion options

Once an experiment is defined the process of disambiguation is manual. The application retrieves from WordNet the different senses of the query terms. The user has to select the correct and/or incorrect meanings of each selected word in the context of a topic. In figure 2 we see the three fields of a topic and a list of the words selected from the title with their meanings and synonym sets. Once the user selects the meanings, the query can be formulated.

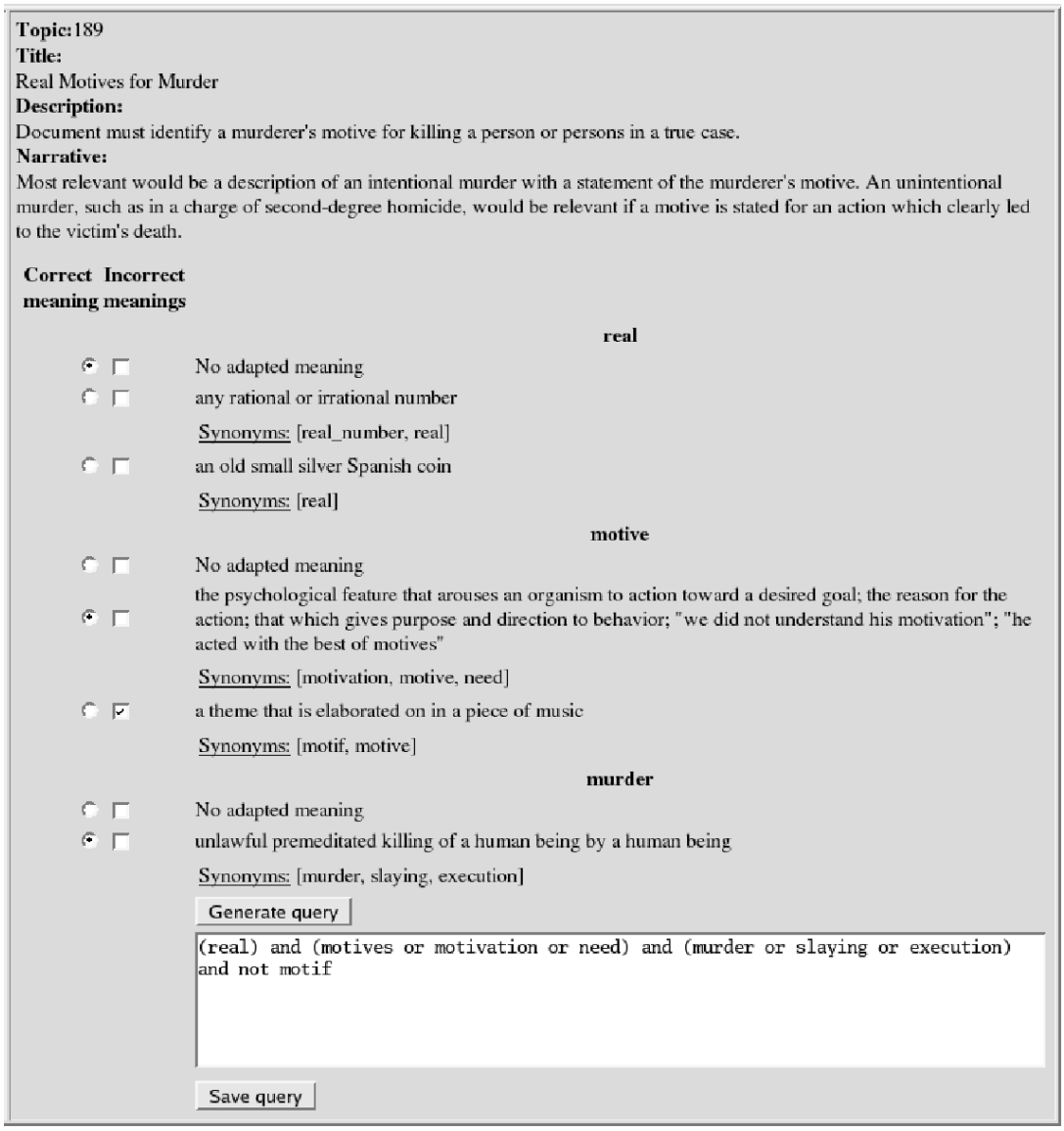

Figure 2. Interface of the application which allows the sense disambiguation and query formulation

5. EXPERIMENTS AND RESULTS

The experiments were done with a subset of the TIPSTER/TREC collection consisting of approximately 173.000 documents. Specifically, we used the Wall Street Journal documents in TIPSTER/TREC volumes 1 & 2. TREC topics #151 to #200 were used for generating the initial queries.

The objective of the first experiment was to test the best connective for expansion. In this experiment the results of the expansion with OR were better than the results of expansion with AND. For the OR expansion the non-interpolated average precision was of 15.66% whereas the AND expansion yields 12,86% average

precision. The T-test shows that the difference is statistically significant. The precision-recall graphic is shown in figure 3 (a).

In all experiments we selected only the synonyms for expansion. Although the initial idea was to include also hyponyms, the nature of hyponyms in WordNet is inadequate for expansion. Usually there are too many to include all without a lot of noise. In most cases only one of the hyponyms is equivalent in the context to the original term. For these reasons a manual selection would be necessary to achieve some positive result. This selection is very difficult because there are too many hyponyms for a word but the number of appropriate hyponyms is small. In the case of synonyms this selection can be made manually because it usually includes more words from a more reduced synonym set.

Once selected the OR as the appropriate connective, the next experiment let us to compare positive and negative expansion. We made experiments adding the synonyms only with OR (positive expansion terms); adding synonyms with OR and with the synonyms for the incorrect meanings of the words as negated terms (negative expansion terms) and finally adding only the negative expansion terms. For a query $a \wedge b$ and the correct synonyms a_1 and a_2 for a, b_1 for b, and the synonyms for incorrect meanings a_3 for a, b_2 and b_3 for b, the query with positive and negative expansion will be $(a \vee a_1 \vee a_2) \wedge (b \vee b_1) \wedge \neg a_3 \wedge \neg b_2 \wedge \neg b_3$. The objective of negative expansion is to penalize the documents containing incorrect synonyms. In addition to this automatic expansion we made two experiments restricting manually the number of added terms. In the case of the expansion only with OR, we removed the least significant synonyms for each original word. The same process was made with the negative terms for the negative and positive expansion experiments, removing the inadequate negated terms that could introduce noise. The results can be found on figure 3 (b).

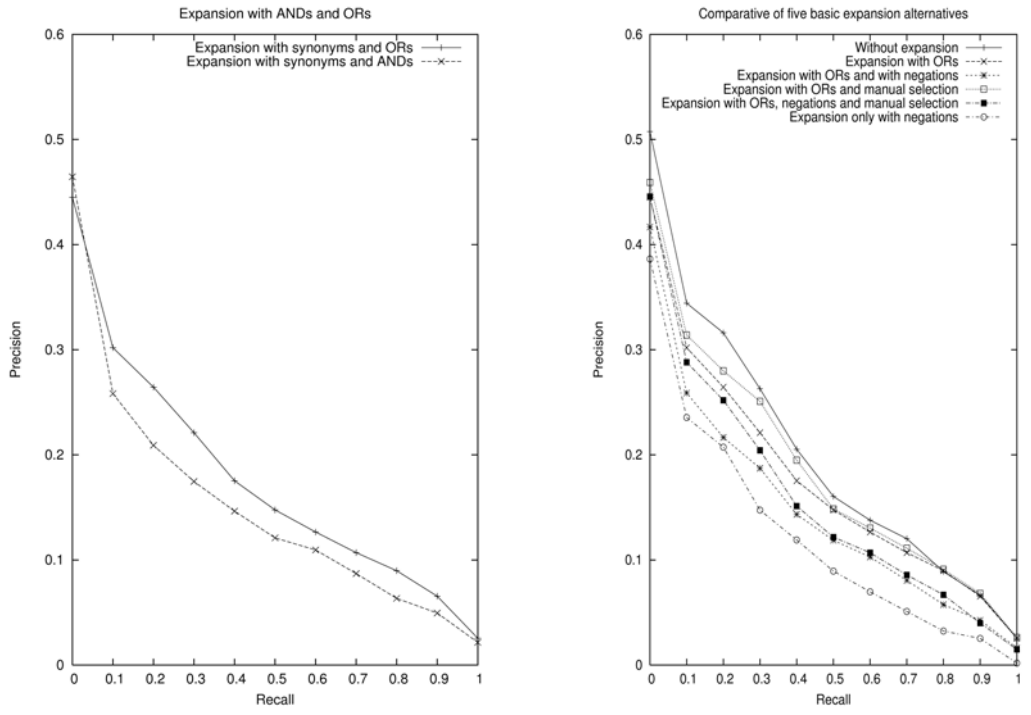

Figure 3. (a) Comparison of OR expansion with AND expansion. (b) Comparison of positive expansion and negative expansion experiments.

Not surprisingly, the results of the experiments with manual selection were better than the results of the experiments with automatic expansion. In any case the precision of the expansion only with OR was larger than the precision in experiments with negative expansion even when manual selection was made. Even in the OR expansion with manual selection the results were not better than the results with no expanded queries, having a 2% decrease in non-interpolated average precision.

The manual selection was made by a user with no experience on information retrieval. We decided to repeat the experiment with expert users. We gave them instructions to be more selective with the number of synonyms to include in the expansion. Under these experimental conditions we got some improvements in precision. Furthermore, in an individual query analysis when the expanded terms kept in the query are more general or more commonly used, the precision was better. In fact, although for the set of fifty topics there is no overall improvement, there was an improvement for 30% of queries.

In an attempt to get automatically a very restricted number of expansion terms, we decide to use the term global frequency to select the expansions terms. In a first experiment, for the set containing the original query word and its synonyms, we select the word with the higher frequency in the test collection. The results of this experiment were worse than those obtained in the case of manual selection. In a second experiment we always kept the original query term and an additional term was selected with the same method. In this case, the results were very similar to those of the case of manual selection having a very small difference in average precision.

Finally, the terms of the original query were selected manually from the topic words instead of selecting it automatically from the title of the topic as it was done in the previous experiments. This manual selection improved the performance of retrieval. The expansion terms used for the selected words were the same of the best previous experiment. The difference of precision, for all standard recall levels, between the results of expanded and not expanded queries experiments was similar to the best experiment.

6. CONCLUSIONS

We have shown in the framework of a logical model of IR the difficulty in the use of WordNet as the only source of linguistic information for query expansion. The basic problem is that the number of synonyms for each word is excessive. Adding all valid synonyms to a query introduces noise that degrades the performance and there is no way, using only WordNet, to select an appropriate subset of synonyms. It would be necessary a measure, inside a synset, with information about the proximity of each word to the meaning, establishing different levels of synonymy, and with the specificity or frequency of use of each word in the language.

Although the different expansion techniques did not improve the results of retrieval, the experimental comparative let us extract some results about the structure in the queries. The results for queries in which the original terms are connected with AND and the expansion terms are connected with OR are better than the results for non-structured queries, i.e., using only the AND connective. The queries expanded with OR have a structure that maintains the importance of each initial query term where each expanded term is selected with its original term. This advances that the expressive framework utilized is suitable and important improvements may be obtained in the future if more efficient ways of handling linguistic information are available. Unfortunately, the use of negative expansion did not contribute to improve the performance of retrieval.

Automatic disambiguation was not the objective of this work, so we made here a manual selection of the correct synset. For each synset we made a manual selection of the most appropriate synonyms for each word in a query. Even with this selection the expansion results were very similar to the baseline results. Nevertheless, an individual query analysis for the best experiment showed that 30% of the queries performed better after expansion. Examining these queries we can say that, for a success in expansion, the expansion terms have to be more general than the original terms and/or more commonly used. At the same time we have to avoid the incorporation of noisy terms in the expanded query.

ACKNOWLEDGEMENTS

The work reported here was co-funded by "Ministerio de Ciencia y Tecnología" and FEDER funds under research projects TIC2002-00947 and Xunta de Galicia under project PGIDT03PXIC10501PN. The third author was supported in part by MCyT and in part by FEDER funds through the "Ramón y Cajal" R&D program.

REFERENCES

1. Dalal, M., 1988. Investigations into a theory of knowledge base revision: preliminary report. *Proceedings of AAAI-88 National Conference on Artificial Intelligence.* Saint Paul, USA, pp. 475-479.
2. Losada, D. E. and Barreiro, A., 2001 A Logical model for information retrieval based on propositional logic and belief revision. *The Computer Journal.* Vol. 44, No. 5, pp. 410-424.
3. Losada, D. E., 2001. A logical model of information retrieval based on propositional logic and belief revision. *PhD thesis*. University of A Coruña.
4. Losada, D. E. and Barreiro, A., 1999. Using a belief revision operator for document ranking in extended boolean models. *Proceedings of ACM SIGIR'99 Conference on Research and Development in Information Retrieval.* Berkeley, USA, pp. 66-73.
5. Losada, D. E. and Barreiro, A., 2000. Implementing document ranking within a logical framework. *Proceedings of SPIRE-2000 Symposium on String Processing and Information Retrieval*. A Coruña, Spain, pp. 188-198.
6. Losada, D. E. and Barreiro, A., 2000. Efficient algorithms for ranking documents represented as DNF formulas. *Proceedings of ACM SIGIR-2000 Workshop on Mathematical and Formal Methods in Information Retrieval.* Athens, Greece, pp. 16-24 .
7. Mandala, R. et al., 1999. Combining multiple evidence from different types of thesaurus. *Proceedings of ACM SIGIR'99 Conference on Research and Development in Information Retrieval*, Berkeley, USA, 1999, pp. 191-197.
8. Miller, G.A. et al., 1990. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, Vol. 3, pp. 235-312.
9. Nie, J.Y., 1998. Using terminological knowledge in information retrieval. *CCAI - The Journal for Integrated Study of Artificial Intelligence, Cognitive Science and Applied Epistemology*, Vol. 15, No. 1-2, pp. 113-144.
10. Nie, J. and Jin, F.,2002. Integrating logical operators in query expansion in vector space model. *Proceedings of ACM SIGIR-2002 Workshop on Mathematical and Formal Methods in Information Retrieval.* Tampere, Findland, pp. 77-88.
11. Qiu and Frei, H.P., 1993. Concept based query expansion. *Proceedings of ACM SIGIR'93 Conference on Research and Development in Information Retrieval*. Pittsburgh, USA, pp. 160-169 .
12. Smeaton, A.F. et al., 1995. TREC-4 Experiments at Dublin City University: Thresholding Posting Lists, Query Expansion with WordNet, and POS Tagging of Spanish. *Proceedings of TREC-4 Conference.* Gaithersburg, USA, pp. 373-390.
13. TREC NIST Web Site: *http://trec.nist.gov*
14. van Rijsbergen, C.J., 1986. A non-classical logic for information retrieval. *The Computer Journal.* Vol. 29, pp. 481-485.
15. Voorhees, E.M., 1994. Query Expansion using lexical-semantical relations. *Proceedings of ACM SIGIR'94 Conference on Research and Development in Information Retrieval.* Dublin, Ireland, pp. 61-69.