Joining automatic query expansion based on thesaurus and word sense disambiguation using WordNet

Francisco João Pinto*
and Antonio Fariña Martínez

Department of Computer Science,
University of A Coruña, Campus de Elviña s/n,
A Coruña, 15071, Spain
E-mail: fjoao@udc.es E-mail: fari@udc.es
*Corresponding author

Carme Fernández Pérez-Sanjulián

Department of Galician-Portuguese, French, and Linguistics,
University of A Coruña,
Campus da Zapateira s/n, A Coruña, 15071, Spain
E-mail: carme@udc.es

Abstract: The selection of the most appropriate sense of an ambiguous word in a certain context is one of the main problems in Information Retrieval (IR). For this task, it is usually necessary to count on a semantic source, that is, linguistic resources like dictionaries, thesaurus, etc. Using a methodology based on simulation under a vector space model, we show that the use of automatic query expansion and disambiguation of the sense of the words permits to improve retrieval effectiveness. As shown in our experiments, query expansion is not able by itself to improve retrieval. However, when it is combined with Word Sense Disambiguation (WSD), that is, when the correct meaning of a word is chosen from among all its possible variations, it leads to effectiveness improvements.

Keywords: automatic query expansion; thesaurus; disambiguation; WordNet.

Reference to this paper should be made as follows: Pinto, F.J., Martínez, A.F. and Pérez-Sanjulián, C.F. (2008) 'Joining automatic query expansion based on thesaurus and word sense disambiguation using WordNet', *Int. J. Computer Applications in Technology*, Vol. 33, No. 4, pp.271–279.

Biographical notes: Francisco João Pinto received his MSc Degree in Mathematics/Computer Science from the Instituto Superior José Varona (Cuba) in 1992. With the support of the Government of Angola he moved to Spain, and in 2008, he obtained his PhD in Computer Science from the University of A Coruña. His research interests focus in information retrieval, and more precisely they include expanding natural language queries by means of word disambiguation, which is of utmost importance to integrate typical web search engines into automated processes.

Antonio Fariña Martínez earned his PhD in Computer Science in 2005 from the University of A Coruña, where he is nowadays an Associate Professor from the Computer Science Department. His research is mainly focused in text retrieval on natural language text collections. Actually, his main contributions to this field are related to text compression, as well as to compressed indexing structures. Other areas of interest are: algorithms, searches in metric spaces, and geographic information systems. He has been member of the Program Committee of several Conferences (MDMM, ICEIS, JIDEE). He has also been external reviewer of various journals (Computer Journal, IJCSSE, JRPIT, …) and international conferences (ESA, CIKM, PSI, VODCA, MICAI).

Carme Fernández Pérez-Sanjulián obtained her PhD in Hispanic Philology (Galician-Portuguese Section) from the University of A Coruña in 2001. She is an Associate Professor of the University of A Coruña. Apart from her interest in topics related to Galician (and Portuguese) Literature, she has been an active researcher in different works related to the use of new technologies for creating and spreading cultural resources, specially focused in the creation of Digital Libraries. In this area she has participated in several research and development projects, such as the creation of the Galician Virtual Library.

1 Introduction

In the last decades the amount of data available in digital form has been growing exponentially. In this scenario, the management of such a large amount of data requires not only dealing with the problems that arise from the big needs of storage media, but also the development of structures that permit the organisation of the data in such a way that the information contained in an information system can be retrieved effectively. Information Retrieval (IR) covers the representation, storage, organisation, and access to the information of such systems. Basically, the representation and organisation of the data must provide an easy form for accessing the bits of information in which a user may be interested (Baeza-Yates and Ribero, 1999). On the one hand, when a user aims at recovering some information from an information system, he has to be able to translate his needs of information into a query that can be processed by a search engine. This query is usually composed of several keywords that cover a user's needs. On the other hand, the main objective of an IR system is to recover that information that may be of interest or *relevant* given a user query.

It is important to notice the differences between the two concepts of data retrieval and information retrieval. Data retrieval consists basically of identifying those documents from among a collection of documents that contain the keywords requested by a user. However, recovering those documents is not enough to satisfy the information needs of the user. In fact, when a user introduces some keywords, it is usually interested in retrieving data regarding some subjects, not only in just retrieving documents, including a set of keywords. Therefore, it is necessary that, given a query, an information retrieval system can try to recover documents containing not only the few terms given by a user, but also an expanded set of terms related to the former ones. Even though there are information retrieval systems that keep well-structured data with well-defined semantics (for example, a relational database), an information retrieval system has to usually deal with non-structured and ambiguous data. For example, this is the scenario of an information retrieval system that deals with natural language text. The former systems can usually answer queries in a very accurate form. In the second case, retrieval could become less precise, and some documents from the retrieved set of documents, might be useless.

An information retrieval system must, somehow, be able to identify the information stored in any document. This is a key property to estimate the relevance of any document with respect to users' queries and, consequently, to permit us to rank the retrieved documents depending on that value. Therefore, an information retrieval system has not only to organise information in such a way that an efficient retrieval can be performed, but it also has to be able to recover all the relevant documents for a query. At the same time, the retrieval system should retrieve as few documents as possible, so that it obtains high precision values.

In Section 2, we introduce both query expansion and Word Sense Disambiguation (WSD), and some previous works that make use of linguistic information to expand queries are discussed. In Section 3, the vector model used is presented. We show how the measurement of the similarity between a query and the documents in a collection can be done. This measurement will be useful to determine the ordering of the relevant documents. WordNet is described briefly in Section 4. In order to perform experiments, an application that permits the formulation of queries was created. It is shown in Section 5, where we also describe how those experiments will be run. Next section presents the experimental results obtained. Finally, the conclusions of our work are discussed in Section 7.

2 Using WordNet in query expansion and word sense disambiguation

Users using a retrieval system that takes the coincidence of words as the base to recover a document face the challenge of expressing their queries with words that are included in the vocabularies of the documents that they are interested in. This is a problematic issue in large text databases because they usually contain many different expressions that refer to the same concept. Nevertheless, the ability to recover documents from these large databases is crucial in many different scenarios; documentation stored in legal databases that are used in a trial, corporative databases, news clipping and the recovery of articles from a digital newspaper, finding relevant passages within a complete set of manuals of a complex system for a particular problem, etc. A way to help a user when some information is requested through the formulation of a query is that the retrieval system can manage to expand the query automatically. That is, that some terms related to the words provided by the user are also searched for during the retrieval process. These new terms that can be statistically related to the original words of the query (for example by focusing in terms that usually occur in the same documents), or can be chosen with the aid of linguistic resources.

The use of statistical information based on the co-occurrence of terms to expand queries is attractive because the relationships between terms can be obtained from the documents in the collection. Therefore, the necessity of using linguistic resources or other tools that can be difficult to maintain can be avoided. Unfortunately, the success of co-occurrence-based methods as a way to improve retrieval effectiveness has been rather poor when no extra relevance data is available. Other techniques like the Latent Semantic Indexing (Deerwester et al., 1990) that are based on the use of statistical relations between terms but do not actually expand queries have proven to obtain good results.

The use of linguistic information as the source of related terms has been applied successfully in some small experiments. In Salton and Lesk (1971) they found that synonym-based query expansion improves the results

obtained. However, when dealing with large collections, the query expansion performed by choosing either more general terms or more specific terms extracted from hierarchical linguistic resources leads to rather inconsistent results. Therefore, such a query expansion process was not useful in practice. Wang et al. (1985) found that a variety of lexical-semantic relationships improves the effectiveness. Unfortunately, those results were obtained in experiments on very small collections, and using linguistic resources from a unique field. Therefore, they have not proven to be successful in wider scenarios.

The most representative works that deal with query expansion based on the use of lexical-semantic relationships presented experiments obtained over large reference collections from TREC,[1] including data from different fields. Among them, Mandala et al. (1999) combined the relationships stored in WordNet (a linguistic resource described in Section 4) with other measures of similarity based on syntactic dependencies and information regarding the co-occurrence of terms within documents. This resulted in an important improvement of the retrieval effectiveness. In Voorhees (1994) the use of WordNet for performing query expansion did not actually increase the effectiveness of the information retrieval process. The expanded queries were evaluated against the topics 101–150 from the second TREC conference (TREC-2). To avoid the effects of a poor selection of words, the terms that constitute the queries to be expanded were generated automatically from the topics. In addition, the sets of synonyms that were considered to be related to such topics were also obtained by using WordNet. For such reasons, the results obtained in that work stand as an upper bound for the effectiveness that can be expected for an expansion method using a fully automatic selection procedure. Even in the best case, such expanded queries did not actually improve the effectiveness obtained with queries that were already complete initially, i.e., the effectiveness of the queries that contained enough representative terms and that included a detailed description of the topic of interest. Nevertheless, when less complete queries (consisting of just a simple sentence describing the topic of interest) were provided, retrieval effectiveness was significantly improved by using query expansion.

The procedure used by Voorhees consists of the following steps: firstly, from the examples of information needs (topics) provided by TREC, an automatic selection of terms is done. Then, also automatically, a set of synonyms (those that were considered to own the most correct meanings) is obtained for each one of those terms by using WordNet. Such synonyms can be:

- direct synonyms
- descendants from an *is-a* WordNet hierarchy
- words from any set of synonyms within distance equal to 1 with respect to the original set of synonyms independent of the type of connection between them
- etc.

In addition, the expansion procedure permitted using different parameters to facilitate the comparison of the effectiveness of such schemes. For a given experiment, and for each relationship included in WordNet, those parameters permit to specify the maximum length of a chain of synonyms. That is, a chain of relationships between terms that arise when the synonyms of a term t are obtained from the synonyms t_{ij} of other terms t_i that are synonyms of t, as well as from the terms t_{ijk} that are synonyms of t_{ij}, and so on. Next, all the synonyms included in a set of synonyms inside such linked sets are added to the query. All the stopwords, that is, very common terms that occur in a language (articles, prepositions, conjunctions, …) are removed from the query. Finally, a stemming process is applied to words in the expanded query, obtaining the morphological root of each word (stem).

In Voorhees's expansion model the stems added through different lexical relationships (either from hierarchical synonyms or from direct synonyms) are kept separated using the extended vector-model introduced by Fox (1983). A query is represented as a vector. Each query-vector consists of subvectors of different types from concepts (ctipos), and the ctipos correspond to different lexical relationships. A query vector can include up to 11 ctipos: the first one is associated with the original terms of the query. Another one is associated with its direct synonyms, and the remaining nine ctipos are associated with the other types of relationships contained within the hierarchy of nouns of WordNet. A term from the original query that is also a member of a selected set of synonyms appears in the two associated ctipos. In a similar way, a word that is related to a set of synonyms through two different connections will appear in both ctipos.

Assuming a vector model, the similarity between a document vector D and an expanded query vector Q was computed as the weighed sum of similarities between D and each subvector from Q. Authors used the same weighting scheme as in the tool *SMART* (Salton and Lesk, 1971) and the *inc* weights suggested in Buckley et al. (1993) to weigh the terms of the vectors. That is, the weight of a term in a document is set to $1.0 + \ln(tf)$, where tf is the number of times that the term occurs in the document. Then normalised values are obtained by means of the square root of the sum of the squares of the weights in the vector (cosine normalisation). The terms of the query are weighed using the logarithm of the frequency factor of the terms and multiplying it by the inverse frequency in the documents and in the terms in ctipo, that represent the original query normalised by means of the cosine. The weights in the additional ctipos are normalised by using the length calculated for the ctipo of the original terms. This normalisation strategy allows the weights of the terms of the original query not being affected by the expansion process and keeps the weights in each ctipo comparable with those from other ctipos.

Voorhees used four different strategies for query expansion:

- expansion based on direct synonyms
- expansion including both direct synonyms and all the descendants in the *is-a* hierarchy
- expansion using direct synonyms, their parents, and all the descendants in the is-a hierarchy
- expansion using both direct synonyms and all the synonyms directly related to the set of direct synonyms (that is, those sets related to the direct set of synonyms by following a chain of length equal to one through any connection among the nine different types of existing connections).

In practice, the weight associated with the subvector of original terms was usually greater than the weight of the expanded subvectors. This was done in order to reflect the assumption that the terms provided by a user are usually more important than those added automatically. The experiments in which the weight of the original terms was smaller or equal than the other weights proved this assumption since worse results were obtained.

The expansion approach proposed in that work was not clearly effective, since none of the expansion strategies improved significantly the results yielded by non-expanded queries. In fact, differences between different runs of expanded and not-expanded individual queries were actually very small. The results obtained for individual queries varied more for the more aggressive expansion strategies. That is, those using long chains of connections and those giving more weight to the added terms. However, for a given set of queries, the global results became worse for the queries expanded in such an aggressive way.

In an initial set of experiments, better results were obtained when queries were expanded by using the set of direct synonyms, and weighting with 0.5 all the added subvectors. For this reason, we will use this query expansion strategy in our experiments in Section 6.

In order to prove the hypothesis that the expansion of terms is not helpful in TREC collections because the description of the problem provided by a TREC topic is very complete, queries obtained as shorter versions of those topics were expanded using standard expansion strategies (automatic expansion with WordNet). In the case of the expanded short queries significant improvements with respect to the non-expanded short ones were obtained. However, as shown previously no improvements were obtained with respect the original ones in its longer version.

To sum up, Voorhees' experiments showed that query expansion based on lexical-semantic relationships leads to very small improvements when the user provides a detailed query. That is, an expanded query is unlikely to be better than a well-formulated query provided by a user. However, as a user does not usually provide a detailed query, this query expansion approach can potentially improve the results obtained with the original query.

In Smeaton and Quigley (1996) queries from TREC-4 collection were expanded with several strategies using weighted expansion of terms, combined with automatic and manual techniques for word meaning disambiguation. Unfortunately, the results were not successful and retrieval worsened in comparison with the non-expanded queries.

In Qiu and Frei (1993) a linguistic resource created automatically was used, improving retrieval effectiveness in around 20% when working over some small collections of text. Despite handmade linguistic resources, their linguistic resource was built automatically without any kind of human interaction.

In Schutze and Peterson (1994) they created a linguistic resource based on the co-occurrence of terms, yielding slightly improved retrieval results over a reduced version of a TREC collection.

In Peat and Willet (1991) the authors provided theoretical results showing the limitations of the data based on the co-occurrence of terms as the base for performing query expansion. Consequently, some researchers have tried to construct linguistic resources using methods with a larger linguistic base. In Grefenstette (1992) a linguistic resource using the syntactic context was created and experiments using several small test collections were performed. Even though the results improved in some small collections, they failed in others. In Jing and Croft (1994) they also obtained some improvements by performing query expansion using linguistic resources constructed automatically and based on grammars.

As shown before, another interesting work is that of Mandala et al. (1999). It is focused on automatic query expansion in addition to the use of co-occurrence information. Query expansion can include all the terms in the relevant documents or some subset among them. In this work, query expansion was carried out by using linguistic resources. Experiments showed significant improvements with respect to the original queries. However, in addition to the combination of several linguistic resources, these improvements are mainly the result of using information related to the co-occurrence of words in relevant documents. Unfortunately, this is not useful in practice, due to its high computational cost.

3 Vector space model

The vector space model of information retrieval, also known as the vector model, was proposed in Salton and Lesk (1971) and it is widely used in information retrieval nowadays. It proposes a framework in which a partial matching is possible by assigning non-binary weights to the terms that can occur in the queries and in the documents. The weights of the terms are used to compute the degree of similarity between any document and a user query.

Let us assume that each document and each query is represented by a vector with the frequency of the terms, that is, $\vec{d} = (x_1, x_2, \ldots, x_n)$ and $\vec{q} = (y_1, y_2, \ldots, y_n)$, respectively, where n is the total number of terms (or size of vocabulary) and x_i and y_i are the frequencies of the term t_i in d and q, respectively. Given a collection C, the *Inverse Document Frequency* (*idf*) of a term is given by $\log(N/n_i)$,

where N is the number of documents in C and n_i is the number of documents that contain the ith term. All the terms in a query and those in a document are weighted with the heuristic equation TFIDF. That is, the weighted d and q vectors are obtained as follows:

$$\vec{d} = (tfd(x_1)\, idf(t_1), tfd(x_2)\, idf(t_2), \ldots, tfd(x_n)\, idf(t_n))$$

$$\vec{q} = (tfd(y_1)\, idf(t_1), tfd(y_2)\, idf(t_2), \ldots, tfd(y_n)\, idf(t_n)).$$

$RawTF = TF(ti, dj)$: indicates the frequency of the term t_i in document d_j.

As shown, $idf = \log(N/n_i)$ is the *idf*. *Idf* depends on the collection. If term t_i appears in all the documents, then $idf = 0$. When t_i occurs only in one document then *idf* gets its maximum value ($idf = \log(N)$).

Finally, the weighted *tf*/*idf* scheme assigns weights to term t_i in document d as follows:

$$W_{i,d} = RawTF \times idf = RawTF \times \log(N / n_i).$$

The function *RawTF* for a query is defined in a similar form as follows (notice that, in this case, *RawTF* depends on the queries rather than on the documents):

$$W_{i,q} = RawTF \times idf = RawTF \times \log(N / n_i).$$

Finally, the score obtained for a document d and a given query q is computed as:

$$S(\vec{d}, \vec{q}) = \sum_{i=1}^{n} tfd(x_i)\, tfd(y_i)\, tfd(t_i). \quad (1)$$

By using equation (1), all the documents in a collection are assigned a score for a given query. Therefore, such documents can be decreasingly ordered by score. This permits the ranking the documents depending on their relevance with respect to that query.

4 WordNet

WordNet is a Machine Readable Dictionary (MRD) for English (Miller et al., 1990). It has become one of the most valuable resources for Natural Language Processing (NLP). WordNet has a database that groups the words into sets of synonyms called *synsets* and provides definitions, comments, examples of use of these words, and the actual meaning in each case. Therefore, it combines the elements of a dictionary (definitions and some examples) with those of a thesaurus (synonyms), resulting in an important support for the automatic analysis of text and words.

Nowadays, the database included in WordNet 2.1 contains around 155,327 words that are organised into more than 117,597 synsets. They form more than 207,016 definitions and senses of the words. WordNet handles four different lexical categories in its synsets (types of elements that can occur in a sentence): nouns, verbs, adjectives, and adverbs. Each synset contains a group of synonyms.

5 Query expansion and disambiguation

The proposed procedure used to perform query expansion consists of the following steps. From the topics in a TREC collection, an automatic selection of terms is done, obtaining the original queries that will be given to the retrieval system. For each one of those terms a set of synonyms is obtained and added to the original query vector. Such sets of synonyms contain the most related synonyms with respect to the terms in the query. This process is also performed automatically with the help of WordNet. The next phase removes the stopwords from the query. Finally, a stemming process is applied to all the remaining words. All these steps are accomplished by an application implemented in Visual Basic.Net. As is shown in Figure 1, this application permits us to choose from among the several options available for performing query expansion.

When experiments are being planned, several choices can be made. Those options are described next.

Firstly, the range of topics (from TREC) that will be used during the experiments should be indicated. For this, the first field in the form in Figure 1 (introducing topics (151–550)) a range is provided using the notation min–max, where min is the number of the starting topic and max the number of last topic to use in the exercise.

Then, a stemming algorithm is selected ("To select stemmer for query" in Figure 1). It is possible either not to use stemming or to apply Porter's algorithm (Porter, 1997). If no stemming is chosen, the complete words in the queries will be used in their normal form. However, if Porter's algorithm is chosen, only the stems (roots) of the words obtained with Porter's algorithm will appear during the retrieval process.

The following option permits us to choose between using stemming and not stemming for accessing WordNet. That is, it allows us to indicate if we want to look for a word in WordNet in an exact way (so that it is only found if its exact form appears in WordNet) or if we are interested in applying the morphologic processing of WordNet. In this way, we will find the words whose root appears in WordNet although it does not appear exactly as introduced.

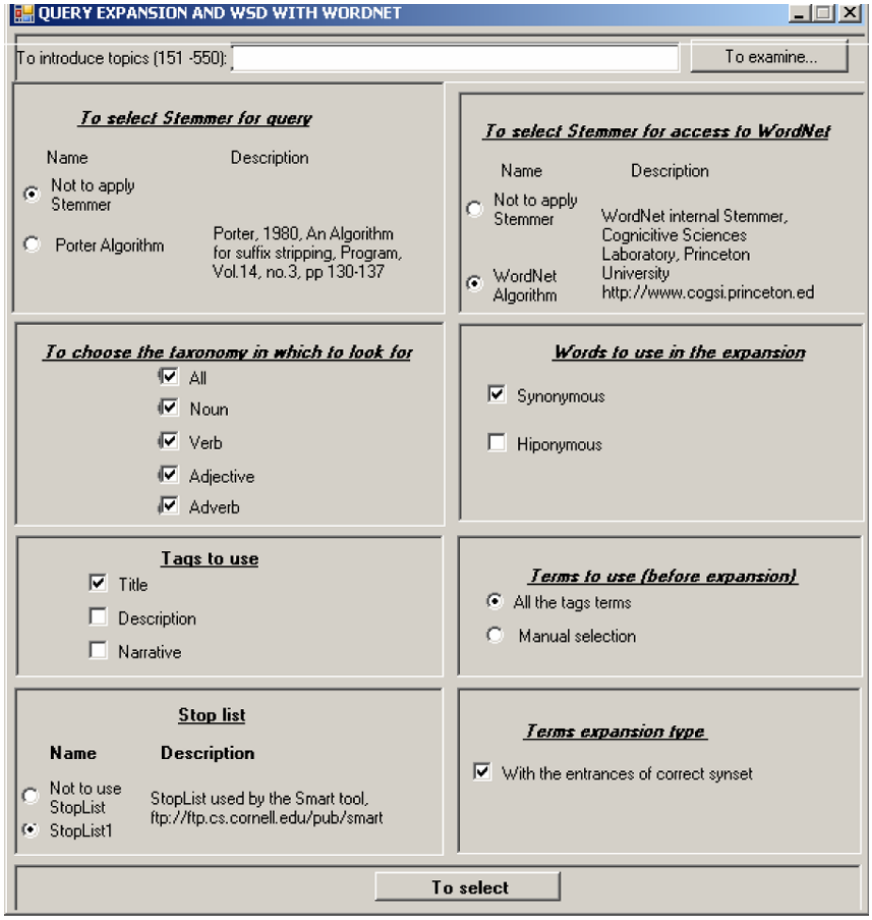
We can also choose the taxonomy in which the meaning of a searched term will be looked for. It is possible to look for words in the taxonomy of nouns, verbs, adjectives, adverbs, or in all the taxonomies, respectively.

The next step permits us to configure how the expansion will be performed. It is possible to use synonyms, first level hyponyms, or both.

As queries are generated from the topics of the titles, the next option permits us to configure which tags to use. Basically, it is possible to choose from among the titles, the description, the text, or any combination.

The next option permits us to choose those terms from the topics which will be used to generate a query (the query without any kind of expansion). It is possible to make a manual selection of terms or to select all the possible terms automatically.

Figure 1 Interface used to choose query expansion parameters (see online version for colours)



It is also possible to apply stopword filtering. In this case, the selected terms are filtered out and very common words are removed.

The last option permits us to indicate the type of query expansion to be performed. It is possible to expand a query with the terms from the synsets chosen as correct ones for the query.

Once the form has been filled, all the parameters that are needed for running an experiment are available. When an experiment begins, all the chosen topics are processed with the same selection of parameters. For each topic, its information (number, title, description, and text) is shown. This is done independently of the type of selection that takes place (either automatically or manually). Moreover, during this process, the meanings of the words selected in the context of the topic are also shown. Then, a list with the chosen words, as well as their possible meanings and their synonyms obtained from WordNet, are also shown. Finally, if a user-driven selection of terms is used, the user will be able to choose the correct meanings for each word that should be used during query expansion.

6 Experimental results

We used an information retrieval system called *lemur* (http://www-2.cs.cmu.edu/~lemur/) and a subset of documents from the reference TREC-8 text collection for evaluating the effects of query expansion in our experiments. More precisely, in all our experiments, topics numbered between 401 and 450 were taken, so that a collection composed of 50 test queries was used in each experiment. The Small Web (WT2g) was the collection of documents used from TREC-8. WT2g contains around 250,000 documents. In addition, the measurements of recall and precision were used to evaluate the results obtained in the retrieval process.

The main goal of our experiments was to investigate whether the expansion of the original queries and the selection of the correct meaning lead to some improvements in the effectiveness obtained during the information retrieval process.

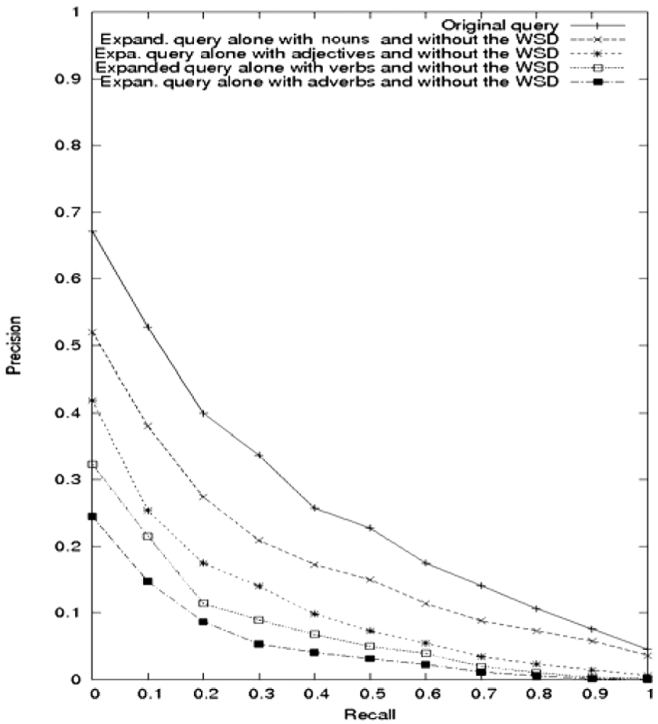
Our experiments consisted in the automatic selection of terms related to each word in the original query, followed by two different types of expansions: the simpler one obtains all the synonyms of each term (but ambiguous words can occur), whereas the second one uses synonyms as the way to select the correct meaning of a term in a given context (with disambiguation of the meaning). The related terms are added to the original query along with the original terms instead of as independent terms added at the end of the query. That is, given a query $\vec{q} = (A, B, C)$, and assuming that terms A' and B' are related to A and B, respectively, the expanded query will be: (A, A', B, B', C).

Once an expanded query is obtained, the title (a field from the query) is indexed using the standard routines of *lemur*, and a ranking of documents is obtained for the original query. This ranking is created by ordering documents in decreasing order of similarity with respect to the original query. Analogously, a ranking of documents can be obtained for an expanded query q' as well as for a disambiguated expanded query q''. From the ranking of documents of each query, the values of recall and precision can be obtained. These values permit us to compare the effectiveness obtained for the different queries q, q', and q'' in our experiments.

6.1 Query expansion without disambiguation

In this section, we focus on the effects of using query expansion in an information retrieval system. In Figure 2, we present the values of recall and precision obtained by the original query and those obtained by applying query expansion of nouns, adjectives, verbs, or adverbs. In these experiments, WSD was not applied. As expected, the best results were obtained when query expansion was performed only over nouns, since it leads to higher values of precision for all the given levels of recall. To sum up, the average precision (non-interpolated) value can be obtained. In practice, when query expansion of the nouns is applied, that value is 16.49%. The values obtained when query expansion is applied over adjectives, verbs, and adverbs are 9.64%, 6.67%, and 4.47% respectively. Finally, the average precision (non-interpolated) for the original query is 24.31% and, consequently, the results obtained by the original query are clearly better than those obtained with the expanded queries.

Figure 2 Recall-precision for the expanded queries without WSD



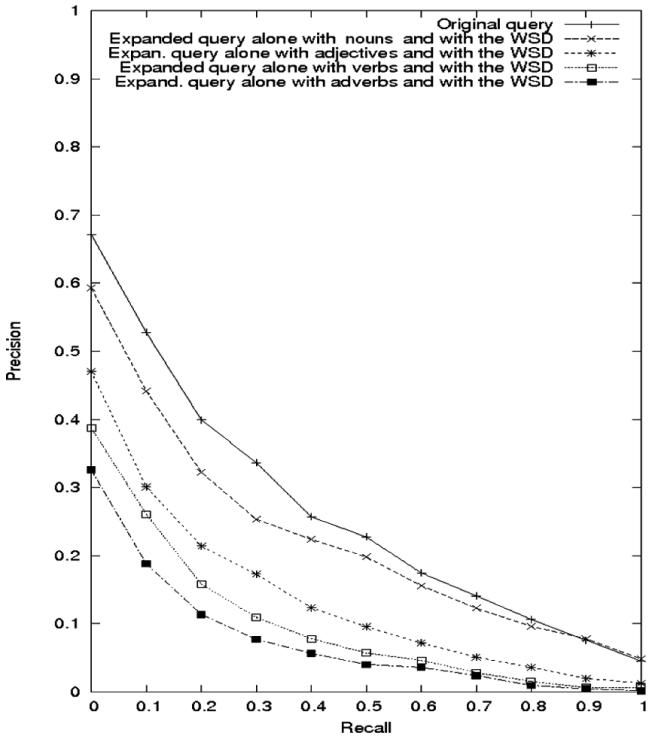
6.2 Query expansion with disambiguation of words

Our dictionary-based disambiguation method makes use of WordNet. Basically, it is based on Pinto (2007a, 2007b) and contains the following steps:

- extraction from WordNet of the sets of synonyms for all the words being disambiguated
- determining the coincidence between the context of the words being disambiguated and the sets of synonyms
- selection of the correct meaning for the words in a text, where the context is given in an automatic form.

In Figure 3, we present the results obtained when query expansion is combined with WSD. By comparing these results with those presented in Figure 2, it can be seen that the use of WSD leads to better effectiveness values for all the syntactic categories over which query expansion was performed. More precisely, by including the disambiguation of nouns into the query expansion process, the average precision (non-interpolated) obtained is around 21%. This new value overcomes that obtained when no disambiguation is used (16.49%). Similar gaps are found with respect to the other existing taxonomies in WordNet.

Figure 3 Recall-precision for the expanded queries with WSD



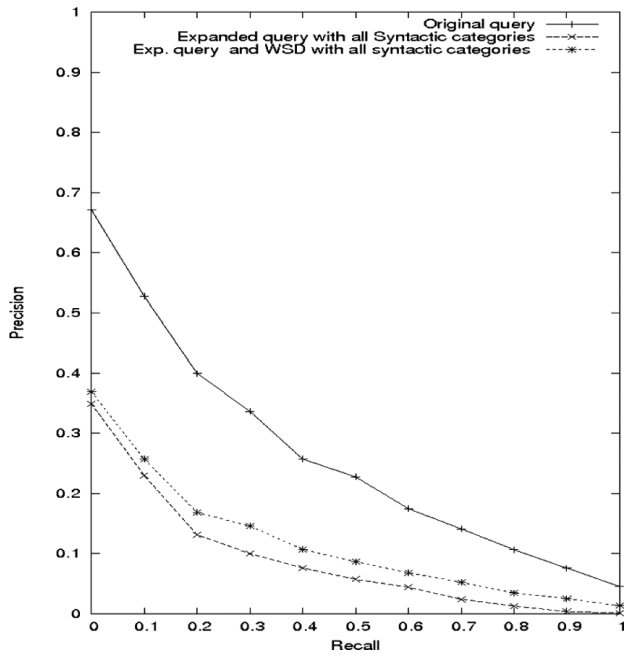
6.3 Full query expansion with disambiguation of words

After showing the advantages of disambiguation in conjunction with query expansion over different syntactic categories, we present the results showing the effects of full query expansion. That is, performing the expansion of all the terms in a query independently of their syntactic category.

Figure 4 shows the values of precision and recall obtained for the original query and those obtained when all the terms in the original query are expanded either with or without WSD. As in the previous sections query expansion (specially with WSD) improves the effectiveness of the original query. However, in comparison with Figure 3, we can see that full query expansion obtains worse results than noun-based query expansion.

In practice, full query expansion yields an average precision (non-interpolated) of 7.67% when WSD is not applied, whereas it obtains 10.41% when WSD is used. Those results are clearly worse than those in Section 6.2 (around 17% and 21%, respectively).

Figure 4 Recall-precision with full query expansion and WSD



7 Conclusions

The WSD is a non-trivial task within an information retrieval system. Many systems try to select the most appropriate sense for a polysemous word using statistics and/or automatic learning. Despite such systems, our proposal uses dictionary-based disambiguation. It uses disambiguation during the query expansion process, yielding interesting improvements in the effectiveness obtained. In our experiments (using vector space model), the new approach yielded improved effectiveness when disambiguation was applied over the different syntactic categories in WordNet. In practice, the best results were obtained when disambiguation was applied during the expansion of nouns. In this case, the new approach obtained average precision (non-interpolated) values of around 21% whereas the previous technique (using query expansion without word disambiguation) could only obtain around 17%.

The results obtained with respect to both the expanded queries and the WSD permit us to extract some interesting conclusions: by using different expansion techniques the results had slightly improved with respect to the original query. Experiments show that by using WSD more advantages from the query expansion with WordNet can be exploited. In practice, our results in a large text collection, support the results obtained in previous research works that reported good effectiveness values when query expansion was used in conjunction with WSD over small text collections.

References

Baeza-Yates, R.A. and Ribeiro-Neto, B. (1999) *Modern Information Retrieval*, Addison Wesley, Boston, MA.

Buckley, C., Salton, G. and Allan, J. (1993) 'Automatic retrieval with locality information using SMART', *Proceedings of the First Text retrieval Conference (TREC-1)*, Nist, pp.59–72.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landewer, T.K. and Harshman, R. (1990) 'Indexing by latent semantic analysis', *Journal of the American Society of Information Science*, Vol. 41, No. 6, pp.391–407.

Fox, E.A. (1983) *Extending the Boolean and Vector Space Models of Information Retrieval with P-norm Queries and Multiple Concept Types*, PhD Thesis.

Grefenstette, G. (1992) 'Use of syntactic context to produce term association lists for text retrieval', *Proceedings of the 15th ACM-SIGIR Conference*, pp.89–97.

Jing, Y. and Croft, B. (1994) 'An association thesaurus for information retrieval', *Proceedings of RIAO*, pp.146–160.

Mandala, R., Tokunaga, T. and Tanaka, H. (1999) 'Combining multiple evidence from different types of thesaurus', *Proceedings of the 22th ACM-SIGIR Conference*, pp.191–197.

Miller, A., Beckwith, R., Fellbaum, C., Gross, D. and Miller, K.J. (1990) 'Introduction to WordNet: an on-line lexical database', *International Journal of Lexicography*, Vol. 3, No. 4, pp.235–312.

Peat, J.H. and Willet, P. (1991) 'The limitations of term coocurrence data for query expansion in document retrieval systems', *Journal of American Society for Information Science*, pp.378–383.

Pinto, F.J. (2007a) 'Uso del Recurso Linguístico WordNet en la Expansión de Consultas con un Modelo de Usuario de Recuperación de Información', *Proceedings of the 2nd CEDI*, In Spanish, pp.45–51.

Pinto, F.J. (2007b) 'Evaluación del Sistema de Recuperación de Información Lemur con Distintos Tipos de Indexación Automática', *Proceedings of the 12th Conferencia de la AEPIA (Zoco'07/CAEPIA)*, In Spanish, pp.21–34.

Porter, M.F. (1997) 'An algorithm for suffix stripping', *Readings in Information Retrieval*, Morgan Kauffmann Publishers, pp.313–316.

Qiu, Y. and Frei, H. (1993) 'Concept-based query expansion', *Proceedings of the 16th ACM-SIGIR Conference*, pp.160–169.

Salton, G. and Lesk, M.E. (1971) *Computer Evaluation of Indexing and Text Processing, the Smart Retrieval System – Experiments in Automatic Document Processing*, Prentice-Hall, Englewood Cliffs, New Jersey, USA, pp.143–180.

Schutze, H. and Pederson, J.O. (1994) 'A coocurrence-based thesaurus and two applications to information retrieval', *Proceedings of the RIAO Conference*, pp.266–274.

Smeaton, A.F. and Quigley, A. (1996) 'Experiments on using semantic distances between words in image caption retrieval', *Proceedings of the 19th International Conference on Research and Development in Information Retrieval*, pp.174–180.

Voorhees, E.M. (1994) 'Query expansion using lexical-semantic relations', *Proceedings of the 17th ACM-SIGIR Conference*, pp.61–69.

Wang, Y., Vandendorpe, J. and Evens, M. (1985) 'Relational thesauri in information retrieval', *Journal of the American Society for Information Science*, Vol. 36, No. 1, pp.15–27.

Note

[1]Text REtrieval Conference, http://www.trec.nist.gov

Website

The LEMUR Project, at http://www-2.cs.cmu.edu/~lemur/