Reformulation of Queries Using Similarity Thesauri

Ángel F. Zazo, Carlos G. Figuerola,
José L. Alonso Berrocal, Emilio Rodriguez

REINA Research Group
University of Salamanca, C/ Francisco de Vitoria 6-16
37008 Salamanca, SPAIN
http://reina.usal.es

Abstract

One of the major problems in information retrieval is the formulation of queries on the part of the user. This entails specifying a set of words or terms that express their informational need. However, it is well-known that two people can assign different terms to refer to the same concepts. The techniques that attempt to reduce this problem as much as possible generally start from a first search, and then study how the initial query can be modified to obtain better results. In general, the construction of the new query involves expanding the terms of the initial query and recalculating the importance of each term in the expanded query. Depending on the technique used to formulate the new query several strategies are distinguished. These strategies are based on the idea that if two terms are similar (with respect to any criterion), the documents in which both terms appear frequently will also be related. The technique we used in this study is known as query expansion using similarity thesauri.

Key words: information retrieval, query expansion, similarity thesaurus, term weighting

1 Introduction

In Information Retrieval (IR) one of the main problems facing users is that of expressing their informational need in a query suitable for the retrieval system. Apart from the requirements of the system for formulating the query, the main problem lies in determining the set of words or terms that express this need semantically. The problem is aggravated owing to the effect of inconsistency in the subjective assigning of terms to concepts, or what is the same, that two people use different words for defining the same concepts (Furnas et al., 1987).

In current IR systems with complete text this approach is especially important. The content of a document is represented using the words that appear in it. The set of index terms is formed by all the words in the document collection after pre-processing the text, which usually includes the elimination of stop words, stemming, selection of terms by semantic category, etc. The user's informational need must also be represented using these terms. The main drawback to this approach is the well-known *word-matching problem*. On the one hand, there are different words that express the same concepts (synonymy) and on the other hand, the same word can have several meanings (polysemy). In these systems it is assumed that two documents are similar if they contain the same words. But this is the problem: the same concept can be expressed with different words, and one same word can appear in documents dealing with completely different subjects.

In this situation, depending on their experience, users will normally have to reformulate their queries until they obtain suitable results. The user begins by sending an initial query to the retrieval system which returns a set of documents ordered according to some relevance criterion of the system. After examining the documents retrieved, the user has to reformulate the query to obtain more relevant results. The greater the number of terms in the query, the smaller the problem, since it will surely include a greater number of index terms that represent relevant documents (Xu and Croft, 2000). In general, the construction of the new query involves expanding terms of the initial query and recalculating the importance of each term in the expanded query (*reweighting of terms*).

In this study we focus on short queries, i.e. queries with one or only a few terms. These queries are particularly interesting because they are the typical ones made on the search engines of the Internet. It is well-known that queries on the web tend to be very short, between one and three terms per query (Jansen et al., 2000; Wolfram et al., 2001). Retrieval results can be improved by expanding the initial query, i.e., including better terms in the query that provide more relevant documents than the original query. The problem lies in finding these better terms.

2 Query Expansion

To expand the query, words or phrases with a similar meaning to those in the initial query must be used. A dictionary or general thesaurus could be used in this process. A thesaurus is a classification system compiled of words and/or phrases organised with the objective of facilitating the expression of ideas. They have been used in retrieval information in the process of creating or expanding queries, i.e. in the process of expressing or broadening the informational needs of a user in a query. Basically this is done by selecting the terms closest to what the user wants. Unfortunately, the use of a general thesaurus for formulating the user's need does not give good results (Voorhees, 1994), mainly because the relations in a general thesaurus are not valid in the local context of user and document collection. Better results are obtained if thesauri or query expansion techniques constructed from the document collection on which the search is launched are used. In any case, reformulation of the query involves two major problems: the choice of the most suitable terms and the weighting of the new terms. This process can be done automatically, i.e., without the intervention of the user, or with the user's collaboration. In the latter case, the user is offered a graphic interface with the results of the search. In (Belkin et al., 2001) the evolution of research in the reformulation of queries in the interactive task of the famous TREC (*Text REtrieval Conference*) is analysed and some user interfaces can be seen. With this interface the users can visualise and select the documents and terms that they consider more relevant to their initial search and the system will use them to reformulate the query.

Mainly three mechanisms are distinguished, depending on the technique used for formulating the new query (Baeza-Yates and Ribeiro-Neto, 1999): (i) the so-called query feedback with the intervention of the user (*user relevance feedback*); (ii) query reformulation, using exclusively information from the documents retrieved (*local analysis*) and (iii) reformulation using overall information from the whole document collection (*global analysis*). These strategies are based on the idea that if two items are similar (with respect to any criterion) the terms that frequently co-occur in them will be related. Based on this idea, semantic relations between the terms are sought. The *Association Hypothesis* (van Rijsbergen, 1979, p. 104) says:

> If an index term is good at discriminating relevant from non-relevant documents then any closely associated index term is also likely to be good at this.

Indeed, considering that the terms that appear in the query are good at discriminating relevant and non-relevant documents, the related terms will also be so. This would allow us to add them to the original query.

The first of the techniques indicated is user relevance feedback. This is a mechanism that has been the object of much study and one of those that give the best results. The process is based on showing the user the results of the first search and after the user has marked the documents they consider relevant and non-relevant, reformulating the query with the terms of these relevant and non-relevant documents. The main advantage is that it is a process guided by the user's criteria of relevance, which is the basis for trying to increase the importance of certain terms and decrease that of others (Salton and Buckley, 1990). One of the algorithms most used is that of Rocchio (Rocchio, 1971). Our research group has also made some experiments on the subject (Figuerola et al., 2002).

The other two techniques for query expansion look for relations or associations between terms: synonyms, derivatives (flexionally and/or derivatively) or proximal ones (which have a minimum distance in number of words). Clustering algorithms are normally used for this. These algorithms are sometimes based on counting the co-occurrences of terms in the documents, and other times on finding similarities under other criteria. Global analysis looks for relations between the terms of the whole document collection, whereas local analysis only considers the documents retrieved and shown to the user for one particular query. Both seek to construct a matrix or thesaurus of relations between the terms, in a global or local sense, and use it to expand the original query with the terms considered best related. Two important strategies stand out in local analysis: on the one hand, what is known as *local clustering*, based on the study by (Attar and Fraenkel, 1977), and, on the other hand, the combination of certain techniques of local and global analysis, called *local contextual analysis* (Xu and Croft, 1996, 2000). Both obtain good results based on the documents retrieved.

The global analysis techniques extract information from the whole collection of documents and use it to expand the query. Research in this field has been developing since before the 1960s. Most of it has had as a basis the study of co-occurrences of terms. It was not until the beginning of the 1990's, however, that satisfactory results were obtained (Peat and Willet, 1991).The main strategies carried out, both automatically and with the intervention of the user, have been based on the simple use of co-occurrence values, the classification of documents or the use of syntactical contexts, with results highly dependent on the document collection to which they were applied, or on parameters that are difficult to calculate (Qiu and Frei, 1993). In contrast with these methods there is another that is attempting to create a similarity thesaurus as a basis for query expansion.

3 Similarity Thesauri

The target here is to construct a similarity thesaurus that makes it possible to expand the complete query (*query concept*), and not only each individual term separately (Qiu and Frei, 1993). A similarity thesaurus is a matrix that is constructed using relations between terms. In its preparation the co-occurrence of terms in the documents is not measured, instead a mechanism is used by which each term in the collection is characterised by the documents in which it appears. This turns the classic concept of information retrieval systems upside down (in these, documents are characterised by index terms, i.e., the terms are used to represent the documents). To construct the similarity thesaurus, the terms of the collection are considered documents, and the documents are used as index terms (they are usually called *index documents*); in other words, the documents can be considered capable of representing the terms.

In order to apply this approach we shall take as a basis the vector model. In the vector model (Salton, 1968) each document d_i in the collection of N documents is represented by a vector $\vec{d_i} = (w_{i1}, w_{i2}, \ldots, w_{im})$, where w_{ij} indicates the weight of the index term t_j in document d_i, and m is the number of index terms in the collection. The weights are normally calculated using the *tf-idf* scheme (Salton and Yang, 1973). The query is also represented in the vector space of terms using a vector, $\vec{q} = (q_1, q_2, \ldots, q_m)$. Each element q_j expresses the degree to which the term t_j represents the informational needs of the person making the query. The main objective of an information retrieval system is to provide information appropriate to the query made by the user. This process depends enormously on the weighting scheme of the terms and subsequent calculation of similarity made by the system. In (Salton and Buckley, 1988) 287 different combinations of assigning weights to terms of documents and queries were experimented with, and we have followed the indications that appear there in order to obtain the best results. The easiest way of obtaining the degree of similarity between a query and a document is to calculate the scalar product of the vectors that represent them (van Rijsbergen, 1979). This is a simple process and one of the most used. So that the similarity value will be between 0 and 1, the vectors of the documents and queries are normalized.

Below we shall se how the similarity thesaurus can be constructed in the vector model. The idea is to turn around the representation model described and instead of considering the documents as represented by index terms, we consider that the index terms can be defined using the documents (*index documents*). Turning the vector model around, each term t_i in the collection of m terms will be represented by a vector of N components in the vector space of documents, $\vec{t_i} = (p_{i1}, p_{i2}, \ldots, p_{iN})$, p_{ij} being a value that expresses the weight of index document d_j in the representation of the term t_i. In order to compute the value of p_{ij} the *tf-idf* scheme is used, but the roles of terms and documents is inverted. The weights are normalized in order to have unit vectors. The calculation proposed in (Qiu and Frei, 1993) is the one indicated in (1). This is the one we used in our experiments.

$$p_{ij} = \frac{(0.5 + 0.5\frac{f_{ij}}{\max_k(f_{ik})}) \cdot itf_j}{\sqrt{\sum_{u=1}^{N}(0.5 + 0.5\frac{f_{iu}}{\max_k(f_{ik})})^2 \cdot itf_u^2}} \tag{1}$$

where

- f_{ij} is the number of times that the term t_i appears in document d_j,
- $max_k(f_{ik})$ is the maximum of the frequency values for the term t_i in the whole document collection (i.e., it will be the value f_{ik}, d_k being the document where it appears most),
- $itf_j = log\frac{m}{|d_j|}$ s the *inverse term frequency* for document d_j, $|d_j|$ being the number of different terms there are in that document.

Calculation of the *inverse term frequency* shows that a short document plays a more important role than a long one. If two terms co-occur in a long document, the probability of them being similar is smaller than if they co-occur in a short one. To calculate the similarity between two terms, t_i and t_j, the scalar product is also used.

$$\mathrm{SIM}(t_i, t_j) = \vec{t_i}^T * \vec{t_j} = \sum_{k=1}^{N} p_{ik} \cdot p_{jk} \tag{2}$$

Calculation for all the terms in the collection produces the similarity thesaurus. It is a symmetric matrix, with values between 0 and 1. Its construction is computationally costly, although it is only done once. If documents are added to the collection the values for the terms included in the new documents must be updated. Once again we must point out that the thesaurus is constructed taking into consideration the representation of terms using index documents, and not co-occurrence.

3.1 Query expansion

The objective when using the similarity thesaurus is to expand the whole query, and not just each individual term separately. For a term to be able to be added to the query there must be a high similarity between this term and all the terms in the query. In the vector space of the terms query q is represented by the vector $\vec{q} = (q_1, q_2, \ldots, q_m)$, where q_i is the weight of term t_i in the query. However, given that the similarity thesaurus has been constructed using the vector space of documents, we must transfer that vector to this space so as to be able to compute the similarity between the whole query and all the index terms in the collection. The term t_i is given by the vector $\vec{t_i}$ in the document space, so that for the whole query, in this space, it will have an importance of $q_i \cdot \vec{t_i}$. We consider that in this space the query only depends on the terms included in it.

Once the query has been represented in the document space, we must obtain the terms most similar to it. We use the measurement of the scalar product with each of the terms t in the collection.

$$\begin{aligned}\mathrm{sim}(q,t) = \vec{q}^T * \vec{t} = (\sum_{t_i \in q} q_i \cdot \vec{t_i})^T * \vec{t} = \sum_{t_i \in q} q_i \cdot (\vec{t_i}^T * \vec{t}) = \\ = \sum_{t_i \in q} q_i \cdot \mathrm{SIM}(t_i, t)\end{aligned} \tag{3}$$

The similarity values between terms are the entries in the similarity thesaurus which have already been calculated. Thus, all the terms of the collection can be put in order according to the value obtained from (3). In general not all the terms are expanded, but only those which have the highest similarity values. We shall denote as r the number of terms that will be added to the original query. The weight, in the space of terms, associated with each term t_e, which will be added to the query, remains to be determined. It seems natural to consider it according to the similarity:

$$q_e = \frac{\mathrm{sim}(q, t_e)}{\sum_{t_i \in q} q_i} \tag{4}$$

The value of the weight of each expanded term is that given in (4). That is, new terms are added to the initial query and the weight of the already existing terms can be modified if they appear among the first r selected for expansion.

4 Experiments

In our experiments we employed the test collection used in several CLEF [1] studies (Peters and Braschler, 2001; Zazo et al., 2002). The characteristics of the collection are given in Table 1. The mean number of terms was considered after eliminating empty words. The document collection came from the news agency EFE, from all the news items of 1994: 215,718 documents (513 MB of information), stored in files, one for each day of 1994. Each file contains several documents, and in these there are fields delimited with SGML labels, as indicated in the example in Figure 1. The relevant fields are `TITLE` and `TEXT`, since the rest contain information such as date, section of the newspaper, etc. The mean number of words per document, considering these fields, is 333.68.

Together with the documents there is a battery of 50 queries in Spanish. Each query is divided into three fields, `ES-title`, `Es-desc` (description) and `ES-narr` (narrative), as can be seen in Fig. 2. Queries can be posed with one, several or all the fields.

[1] CLEF collections are property of CLEF Consortium and ELDA

Table 1: Characteristics of the Collection.

Collection EFE'94	
No. of documents	215,738
No. of queries	50
No. of index terms	352,777
Mean number of words per document	333.68 (max. 2210, min 9)
Mean number of unique index terms per document	120.48

```
<DOC>
<DOCNO>EFE19940101-00002</DOCNO>
<DOCID>EFE19940101-00002</DOCID>
<DATE>19940101</DATE>
<TIME>00.34</TIME>
<SCATE>VAR</SCATE>
<FICHEROS>94F.JPG</FICHEROS>
<DESTINO>ICX MUN EXG</DESTINO>
<CATEGORY>VARIOS</CATEGORY>
<CLAVE>DP2404</CLAVE>
<NUM>100</NUM>
<PRIORIDAD>U</PRIORIDAD>
<TITLE>   IBM-WATSON
          FALLECIO HIJO FUNDADOR EMPRESA DE COMPUTADORAS
</TITLE>
<TEXT> Nueva York, 31 dic (EFE).- Thomas Watson junior, hijo
 del fundador de International Business Machines Corp. (IBM),
 falleció hoy, viernes, en un hospital del estado de Connecticut
 a los 79 años de edad, informó un portavoz de la empresa.
    Watson falleció en el hospital Greenwich a consecuencia de
 complicaciones tras sufrir un ataque cardíaco, añadió la fuente.
    El difunto heredó de su padre una empresa dedicada
 principalmente a la fabricación de máquinas de escribir y la
 transformó en una compañía líder e innovadora en el mercado de
 las computadoras. EFE
    PD/FMR
    01/01/00-34/94
</TEXT>
</DOC>
```

Figure 1: A document from the collection.

In our experiments we considered an index term to be composed of any set of alphanumerical characters, i.e., we also included numbers as index terms. In the lexical pre-processing of the text the terms of the documents and queries that can be used as index terms are obtained. This pre-processing was carried out in several stages. The first consisted of an initial lexical analysis with the objective of determining the processing to be carried out on number, hyphens, punctuation marks, capital and/or small letters, proper names, acronyms, etc. Thus we eliminated all the non-alphanumerical characters, did not detect proper names or acronyms and stored the index

```
<top>
<num> C042 </num>
<ES-title> Naciones Unidas y Estados Unidos invaden Haití</ES-title>
<ES-desc> Encontrar documentos sobre la invasión de Haití por los
soldados de la ONU y de los Estados Unidos. </ES-desc>
<ES-narr> Los documentos comentan tanto la discusión sobre la
decisión de la ONU de enviar las tropas americanas a Haití como la
invasión misma. Se habla también de sus consecuencias
directas.</ES-narr>
</top>
```

Figure 2: A query from the collection.

terms in small letters without considering accents.

The second action performed in pre-processing the text was the elimination of stop words. With the objective of reducing the number of index terms, words that are not very significant in the information retrieval process because of their slight semantic capacity or high frequency are not included. This set of words, known as stop words, is composed of prepositions, articles, adverbs, conjunctions, possessives, demonstratives, pronouns, some verbs and some nouns. With the elimination of stop words, the aim is to reduce the noise that they could introduce in the retrieval. To eliminate stop words, the text was separated into words, which were then checked against the stop word list. In our experiments we used a list of 573 words.

The next step consisted of stemming. Stemming is the process whereby morphological variations of the terms are sought in order to extract the common root. The canonical form of the root represents the variations of the terms deriving from it. In our case, we did not apply stemming.

Next, we selected the terms or groups of terms that would be the index terms. This is normally done according to the syntactical nature of the term, since those that act grammatically as nouns usually have a greater semantic content than verbs, adjectives or adverbs. Our selection of terms was not made on the basis of any syntactical criterion. Simply, those remaining after eliminating the stop words were considered as index terms.

The last step in the pre-processing of terms was the construction or application of a thesaurus that allowed the query to be expanded. Query expansion has already been described. The objective of this study was to show the characteristics of query expansion using similarity thesauri on the EFE'94 document collection.

Once the index terms of the test collection had been obtained, representation of documents and queries was done using the *tf-idf* weighting mechanism and the recommendations of (Salton and Buckley, 1988). We used the scalar product to calculate similarities between documents and queries. In the expansion of each query a local similarity thesaurus was calculated for the terms included in it. The related terms according to (3) were obtained and arranged in decreasing order to then take the first r. The results were evaluated by calculating the average precision for all the queries in the collection in three values representative of recall: a low value of 25%, a medium value of 50% and a high value of 75%. Then the mean of these three values was taken.

In this study we also wished to measure the influence of the number of terms in the original query on the results. For this purpose we did several tests considering different fields of the query: test T (`ES-title` field), test D (`ES-desc` field) and test N (`ES-narr` field). Table 2 shows the improvement in query expansion, with an expansion of 500 terms added to the original query. It can be noted that the greater the number of terms in the original query, the smaller the improvement. This was to be expected, since long queries contain greater description of the

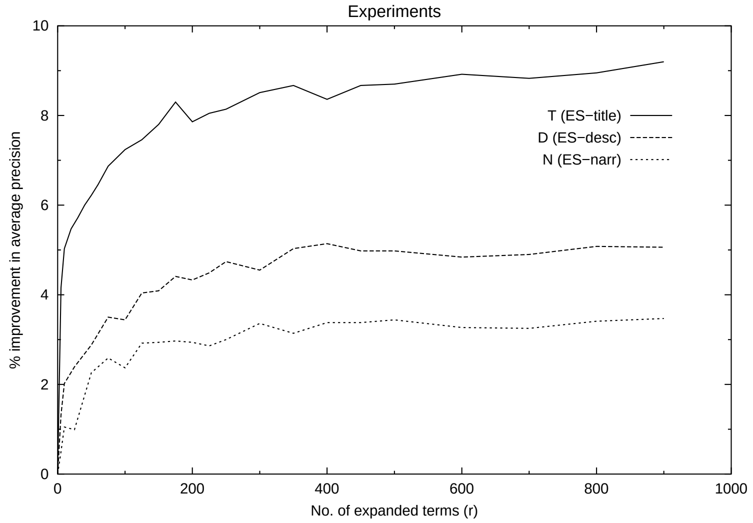

Figure 3: Improvement in the Average Precision according to the number of terms expanded. Tests: T (Es-title field), D (Es-desc field) and N (Es-narr field).

user's informational need than short queries.

Fig. 3 shows the results of our experiments according to the number of terms expanded. We can see that the improvement follows the same course for all three tests. A major improvement is noted with the expansion of a few terms, but after a certain number (around 250) the improvement increases very slowly, and continues to increase, perhaps indefinitely. This seems to coincide with the experiments of other authors (Qiu and Frei, 1993) when the test collection is fairly large, as is our case. This may have to do with the fact that more discriminating terms are needed in large test collections.

5.Conclusions

The technique described obtains good results in query expansion. A thesaurus of similarity between terms was constructed, taking advantage of the possibilities the documents have to represent the terms. This duality in the representation of information allowed us to construct the thesaurus taking into account a weighting scheme according to the length of the documents and not just the number of times that a term appears in that document. The main characteristic resides, however, in the fact that the expanded terms were chosen and weighted taking into consideration the terms of the whole query, and not each individual term separately.

Moreover, as opposed to other techniques of local analysis that base their expansion on the terms of a few documents with user relevance feedback, here we used a technique that related all the terms in the test collection to each other. In fact, the results are worse, mainly because the expansion we used did not incorporate any type of additional relevance information, wich

Table 2: Results of the experiment.

Pruebas	T	D	N
Mean no. of terms per query	2.74	8.40	16.66
Average precision original queries	0.3218	0.3717	0.3634
Average precision expanded queries with r =500	0.3505	0.3902	0.3759
Improvement	8.92 %	4.98 %	3.44 %

other methods do incorporate. It is, therefore, a mechanism that can be executed automatically a sone more characteristic of the information retrieval system. The main drawback lies in the high computational cost required for the construction of the similarity thesaurus. Fro new documents the task is more simple, since the values of the terms appearing in them are modified.

One very interesting aspect is that the smaller the query expansion the better it is. Considering that the queries on the search engines on the Internet usually contain one, two or three terms, this technique can be especially useful in this context. However, considering that the volatility of the information on the Internet is very high, and that major modification of the thesaurus would be required, since the number of documents that would disappear from the collection and the number that would be incorporated is enormous, we believe that this technique can only be used in more static situations.

References

Attar, R. and Fraenkel, A. (1977). Local feedback in a full-text retrieval system. *Journal of the ACM*, 24(3).

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval.* Addison-Wesley, Harlow, England.

Belkin, N., Cool, C., Kelly, D., Lin, S.-J., Park, S., Pérez-Carballo, J., and Sikora, C. (2001). Iterative exploration, design and evaluation of support for query reformulations in interactive information retrieval. *Information Processing & Management*, 37(3):403–434.

Figuerola, C. G., Zazo Rodríguez, Á. F., and Alonso Berrocal, J. L. (2002). La interacción con el usuario en los sistemas de recuperación de información: realimentación por relevancia. *SCIRE, Representación y Organización del Conocimiento.* In press.

Furnas, G. W., Landauer, T. K., Gomez, L. M., and Dumais, S. T. (1987). The vocabulary problem in human-system communication. *Comunications of the ACM*, 30(11):964–971.

Harman, D. K. (1992). Relevance feedback and other query modification techniques. In Frakes, W. B. and Baeza-Yates, R., editors, *Information retrieval: Data Structures and Algorithms*, pages 241–263. Prentice-Hall Inc., Englewood Cliffs (NJ).

Jansen, B. J., Spink, A., and Saracevic, T. (2000). Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Procession & Management*, 36(2):207–227.

Peat, H. J. and Willet, P. (1991). The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the American Society for Information Science*, 42(5):378–383.

Peters, C. and Braschler, M. (2001). Cross-language system evaluation: The CLEF campaigns. *Journal of the American Society for Information Science and Technology*, 53(12):1067–1072.

Qiu, Y. and Frei, H.-P. (1993). Concept-based query expansion. In *Proceedings of the Sixteenth International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 160–169, Pittsburgh (US).

Rocchio, J. (1971). Relevance feedback in information retrieval. In Salton, G., editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. Prentice-Hall Inc., Englewood Cliffs (NJ).

Salton, G. (1968). *Automatic Information Organization and Retrieval.* McGraw-Hill, New-York.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523.

Salton, G. and Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297.

Salton, G. and Yang, C. S. (1973). On the specification of term values in automatic indexing. *Journal of Documentation*, 29(4):351–372.

van Rijsbergen, C. (1979). *Information Retrieval.* Dept. of Computer Science, University of Glasgow, second edition. Also on-line: `http://www.dcs.gla.ac.uk/Keith/`.

Voorhees, E. (1994). Query expansion ussing lexical-semantic relations. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 61–69.

Wolfram, D., Spink, A., Janses, B. J., and Saracevic, T. (2001). Vox populi: The public searching of the web. *Journal of the American Society for Information Science and Technology*, 52(12):1073–1074.

Xu, J. and Croft, W. B. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11.

Xu, J. and Croft, W. B. (2000). Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, 18(1):79–112.

Zazo, Á. F., Figuerola, C. G., Berrocal, J. L. A., and Gómez, R. (2002). Recuperación de información utilizando el modelo vectorial. participación en el taller CLEF-2001. Technical Report DPTOIA-IT-2002-006, Departamento de Informática y Automática - Universidad de Salamanca. On line: `http://tejo.usal.es/inftec/2002/DPTOIA-IT-2002-006.pdf`.