USING A QUERY EXPANSION TECHNIQUE TO IMPROVE DOCUMENT RETRIEVAL

AbdelImgeid Amin Aly

Abstract: Query expansion (QE) is a potentially useful technique to help searchers formulate improved query statements, and ultimately retrieve better search results. The objective of our query expansion technique is to find a suitable additional term. Two query expansion methods are applied in sequence to reformulate the query. Experiments on test collections show that the retrieval effectiveness is considerably higher when the query expansion technique is applied.

1. Introduction

Since the 1940s the problem of Information Retrieval (IR) has attracted increasing attention, especially because of the dramatically growing availability of documents. IR is the process of determining relevant documents from a collection of documents, based on a query presented by the user.

There are many IR systems based on Boolean, vector, and probabilistic models. All of them use their model to describe documents, queries, and algorithms to compute relevance between user's query and documents. Each model contains some constraints, which cause disproportion between expected (relevant) documents and documents returned by IR system. One of the possibilities (how to solve the disproportion) is systems for automatic query expansion, and topic development observing systems. In this respect, query expansion aims to reduce this query/document mismatch by expanding the query using highly "correlated" to the query terms, words or phrases with a similar meaning or some other statistical relation. To detect such correlations between terms, different based-statistical-measures approaches, requiring the analysis of the entire document collection, have been introduced, e.g., term Co-occurrence measures or lexical co-occurrence measures [1, 2]. Query expansion (or term expansion) is the process of supplementing the original query with additional terms, and it can be considered as a method for improving retrieval performance. The method itself is applicable to any situation irrespective of the retrieval technique(s) used. The initial query (as provided by the user) may be an inadequate or incomplete representation of the user's information need, either in itself or in relation to the representation of ideas in documents.

There are three types of QE: manual, automatic, and interactive. Manual QE takes place when the user refines the query by adding or deleting search terms without the assistance of the IR system. New search terms may be identified by reviewing previous retrieval results, communication with librarians or colleagues; other related documents, or a general vocabulary tool are not specific to the IR system (e.g., a dictionary or standard thesaurus) [3]. Decisions about the association of terms are up to the users themselves and are dependent on the expertise of the users with the search system and features [4].

Query expansion involves adding new words and phrases to the existing search terms to generate an expanded query. However, previous query expansion methods have been limited in extracting expansion terms from a subset of documents, but have not exploited the accumulated information on user interactions. We believe that the latter is extremely useful for adapting a search method to the users. In particular, we will be able to find out what queries have been used to retrieve what documents, and from that, to extract strong relationships between query terms and document terms and to use them in query expansion.

Query expansion, as depicted in Figure 1, can be performed manually, automatically or interactively (also known as semi-automatic, user mediated, and user assisted).

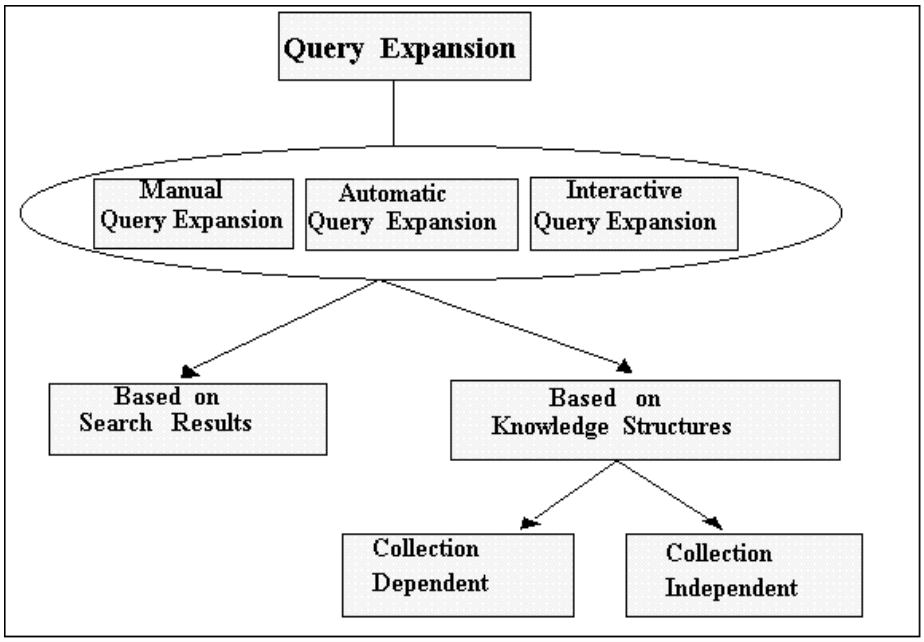

Figure 1: Query Expansion: Methods and Sources

2. Related Works

The existing state-of-the-art query expansion approaches can be classified mainly into two classes: global analysis and local analysis.

Global analysis is one of the first techniques to produce consistent and effective improvements through query expansion. One of the earliest global analysis techniques is term clustering [5], which groups document terms into clusters based on their co-occurrences. Queries are expanded by the terms in the same cluster. Other well-known global techniques include Latent Semantic Indexing [6], similarity thesauri [1], and Phrase Finder [7]. Global analysis requires corpus-wide statistics such as statistics of co-occurrences of pairs of terms, which results in a similarity matrix among terms. To expand a query, terms which are the most similar to the query terms are identified and added. The global analysis techniques are relatively robust; but corpus-wide statistical analysis consumes a considerable amount of computing resources. Moreover, since it only focuses on the document side and does not take into account the query side, global analysis cannot address the term mismatch problem well.

Different from global analysis, local analysis uses only some initially retrieved documents for further query expansion. The idea of local analysis can be traced back at least to a 1977 paper [8]. A well-known local analysis technique is relevance feedback [9,10], which modifies a query based on users' relevance judgments of the retrieved documents. Typically, expansion terms are extracted from the relevant documents. Relevance feedback can achieve very good performance if the users provide sufficient and correct relevance judgments. Unfortunately, in a real search context, users usually are reluctant to provide such relevance feedback information. Therefore, relevance feedback is seldom used by the commercial search engines.

To overcome the difficulty due to the lack of sufficient relevance judgments, pseudo-relevance feedback (also known as blind feedback) is commonly used. Local feedback mimics relevance feedback by assuming the top-

ranked documents to be relevant [11]. Expansion terms are extracted from the top-ranked documents to formulate a new query for a second cycle retrieval.

In recent years, many improvements have been obtained on the basis of local feedback, including re-ranking the retrieved documents using automatically constructed fuzzy Boolean filters [12], clustering the top-ranked documents and removing the singleton clusters [13], clustering the retrieved documents and using the terms that best match the original query for expansion. In addition, recent TREC results show that local feedback approaches are effective and, in some cases, outperform global analysis techniques [14]. Nevertheless, this method has an obvious drawback: if a large fraction of the top-ranked documents is actually irrelevant, then the words added to the query (drawn from these documents) are likely to be unrelated to the topic and as a result, the quality of the retrieval using the expanded query is likely to be worse. Thus the effects of pseudo-feedback strongly depend on the quality of the initial retrieval.

Recently, Xu and Croft [15] proposed a local context analysis method, which combines both local analysis and global analysis. First, noun groups are used as concepts, which are selected according to their co-occurrences with the query terms. Then concepts are chosen from the top-ranked documents, similarly to local feedback.

3. Traditional Document Retrieval

The task of traditional document retrieval is to retrieve documents which are relevant to a given query from a fixed set of documents, i.e. a document database. In a common way to deal with documents as well as queries, they are represented using a set of index terms (simply called terms) by ignoring their positions in documents and queries. Terms are determined based on words of documents in the database, usually during pre-processing phases where some normalization procedures are incorporated (e.g. stemming and stop-word elimination).

3.1 Vector Space Model

The vector-processing model of retrieval is used to transform both the available information requests as well as the stored documents into vectors of the form:

$$D_i = (w_{i1}, w_{i2}, ..., w_{it}) \quad (1)$$

where D_i represents a document (or query) text and w_{ik} is the weight of term T_k in document D_i. A weight of zero is used for terms that are absent from a particular document, and positive weights characterize terms actually assigned. The assumption is that t terms in all are available for the representation of the information.

In choosing a term weighting system, low weights should be assigned to high-frequency terms that occur in many documents of a collection, and high weights to terms that are important in particular documents but unimportant in the remainder of the collection. The weight of terms that occur rarely in a collection is relatively unimportant because such terms contribute little to the needed similarity computation between different texts.

A well-known term weighting system following that prescription assigns weight w_{ik} to term T_k in query Q_i in proportion to the frequency of occurrence of the term in Q_i, and in inverse proportion to the number of documents to which the term is assigned. [16, 17] Such a weighting system is known as a tf x idf (term frequency times inverse document frequency) weighting system. In practice the query lengths, and hence the number of of non-zero term weights assigned to a query, vary widely. To allow a meaningful final retrieval similarity, it is convenient to use a length normalization factor as part of the term weighting formula. A high- quality term weighting formula for w_{ik}, the weight of term T_k in query Q_i is

$$w_{ik} = \frac{(\log(f_{ik})+1.0)*\log(N/n_k)}{\sqrt{\sum_{j=1}^{t}[(\log(f_{ij})+1.0)*\log(N/n_j)]^2}} \quad (2)$$

where f_{ik} is the occurrence frequency of T_k in Q_i, N is the collection size, and n_k the number of documents with term T_k assigned. The factor $\log(N/n_k)$ is an inverse collection frequency ("idf") factor which decreases as terms are used widely in a collection, and the denominator in expression (2) is used for weight normalization.

The weight assigned to terms in *documents* are much the same. In practice, for both effectiveness and efficiency reasons the *idf* factor in the documents is dropped [18, 19]. The term T_k included in a given vector can in principle represent any entities assigned to a document for content identification. Such terms are derived by a text transformation of the following kind: [20]

1. recognize individual text words
2. use stop lists to eliminate unwanted function words
3. perform suffix removal to generate word stems
4. optionally use term grouping methods based on statistical word co-occurrence or word adjacency computations to form term phrases (alternatively syntactic analysis computations can be used)
5. assign term weights to all remaining word stems and /or phrase stems to form the term vector for all information items.

Once term vectors are available for all information items, all subsequent processing is based on term vector manipulations.

The fact that the indexing of both documents and queries is completely automatic means that the results obtained are reasonably collected independently and should be valid across a wide range of collections.

3.1.1 Text Similarity Computation

When the text of document D_i is represented by a vectors of the form (d_{i1}, d_{i2}, …, d_{it}) and query Q_j by the vector (q_{j1} , q_{j2}, …,q_{jt}), a similarity (S) computation between the two items can conveniently be obtained as the inner product between corresponding weighted term vector as follows:

$$S(D_i, Q_j) = \sum_{k=1}^{t}(d_{ik} * q_{jk}) \quad (3)$$

Thus, the similarity between two texts (whether query or document) depends on the weights of coinciding terms in the two vectors.

In the following section we discuss the query expansion technique that will be used for comparison.

4. Query expansion

Query expansion algorithms at first evaluate given query on collection of documents, and then select from relevant documents appropriate terms. The original query is expanded with such selected terms. The expanded query is used to retrieve new set of relevant documents. In this paper we apply two query expansion methods in sequence to reformulate the query so that it will suit to the user's needs more appropriately. One method we

applied is similarity thesaurus based expansion [1], and the other is local feedback method. The similarity thesaurus we use, based on [1], calculates the relevance between terms and queries and is constructed by interchanging the role of documents and terms in retrieval model. The relevance of a term in the similarity thesaurus to the concept of the query is the sum of the weighted relevance of the term to each term in the query. The queries are expanded by adding top *n* relevant terms, which are most similar to the concept of the query, rather than selecting terms that are similar to the query terms.

The local feedback method is similar to traditional relevance feedback method [21], which modifies queries by using the result of the initial retrieval, except that the latter uses the judgment set for calculating re-weighting while the former assumes that the terms in the top ranked *n* documents are relevant to the user's request. Queries are expanded by adding the weight of terms in relevant documents and reducing the weight of terms in last *m* documents of the initial retrieval.

We modify the traditional Rocchio expansion equation to include the query expanded by the thesaurus method and to include negative evidence from the lowest ranked documents rather than non-relevant documents. The new query Q_{new}, including thesaurus expansion, can be defined as the following:

$$Q_{new} = \alpha_1 Q_{org} + \alpha_2 + Q_{te} + \beta \sum_{top} D_i - \gamma \sum_{last} D_j \quad (4)$$

Here, Q_{org} is a initial query, Q_{te} is a query expanded by the similarity thesaurus based method, $\sum_{top} D_i$ represents terms in top ranked documents retrieved in the initial run, and $\sum_{last} D_j$ is terms in low ranked documents. The parameters $\alpha_1, \alpha_2, \beta$ and γ represent the importance of each item. Currently, these parameters are given by human experience. For the initial retrieval, we used the queries expanded by thesaurus method. In this study, we set the parameters as following: $\alpha_1 = 1$, $\alpha_2 = 0.5$, $\beta = 0.6$, and $\gamma = 0.3$.

5. Experiments and their Results

In our experiments, we used the three standard test collections (CISI, NPL, and CACM). We evaluate the performance of the retrieval by average precision measure. Precision is the ratio of the number of relevant documents retrieved to the total number retrieved. The average precision of a query is the average of precisions calculated when a relevant document is found in the rank list. All the query's average precisions are averaged to evaluate an experiment.

Table (1) shows the retrieval quality difference between the original queries and the expanded queries. It seems that the improvement increases with the size of collection.

Table 1: Improvement using expanded queries

Collection	CISI	CACM	NPL
Documents	1035	3205	11430
Avg. precision of original query	0.5547	0.2819	0.1918
Number of additional terms	80	100	800

Avg. precision of expanded query	0.6445	0.3438	0.2448
Improvement	16.19 %	21.96 %	27.63 %

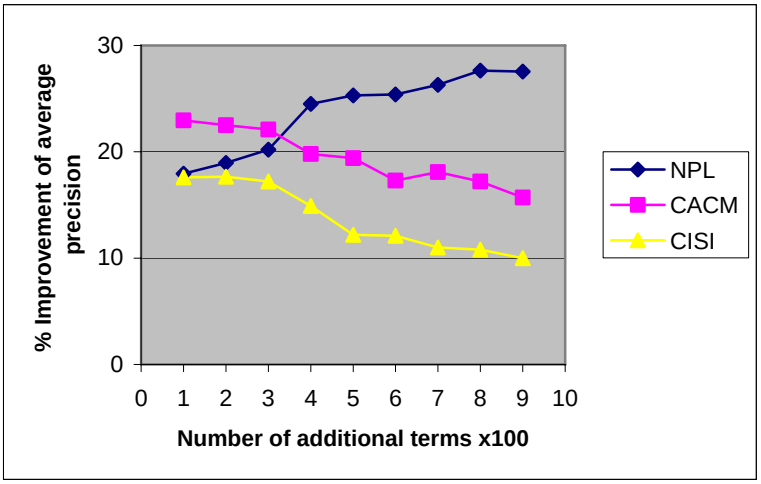

fig.2: Improvement using expanded queries with various numbers of additional terms

The figure indicates that our query expansion technique yields a considerable improvement in the retrieval effectiveness. It seems that the improvement increases with the size of the collection. In addition, the improvement increases with the number of additional search terms that expand the original query as long as the collection is large enough. In Fig. 2, we show how the number of additional terms affects the retrieval effectiveness. It can be seen easily that the improvement by expanded queries increases when the number of additional terms increases. When the number of additional terms is between 100 and 200, the improvement of the retrieval effectiveness remains constant in the small collections CISI and CACM. Once the number of additional terms gets to be larger than 200, the improvement decreases in the small collections, but continues to increase in the relatively large collection NPL. This could be explained by the fact that more search terms are needed to distinguish relevant documents from non-relevant documents in large collections.

6. Conclusion

We presented a two query expansion methods in sequence to reformulate the query. Our experiments made on three standard test collections with different sizes and different document types have shown considerable improvements vs. the original queries in the standard vector space model. Experiments on test collections showed that the improvement increases with the size of the collection. In addition, the improvement increases with the number of additional search terms that expand the original query as long as the collection is large enough. Also it has been pointed out how the number of additional terms affects the retrieval effectiveness.

Bibliography

[1] Y., Qiu and H. P. Frei. Concept based query expansion. In Proceedings of the ACM-SIGIR Intl. Conference on Research and Development in Information Retrieval, pages 160-169, 1993.

[2] E. M., Voorhees. Query expansion using lexical-semantic relations. In Proceedings of the ACM-SIGIR Intl. Conference on Research and Development in Information Retrieval, Dublin, pages (61-70), 1994.

[3] J., Greenberg, Optimal query expansion (QE) processing methods with semantically encoded structured thesauri terminology. Journal of the American Society for Information Science and Technology, 52(6), 487-498, (2001).

[4] E. N., Efthimiadis, Query expansion. Annual Review of Information Science and Technology, 31, 121-187, 1996.

[5] S. K. Jones, Automatic keyword classification for information retrieval, Butterworth's (London), 1971

[6] S. Deerwester, S. T. Dumai, G.W. Furnas, T. K. Landauer, R. Harshman, Indexing by latent semantic analysis, Journal of the American Society For Information Science volume 41(4)pages 391-407,1994

[7] Y. Jing, W. B. Croft, An association thesaurus for information retrieval, RIAO 94 Conference Proceedings pages 146-160,1994

[8] R. Attar, A. S. Fraenkel, Local feedback in full-text retrieval systems, Journal of the ACM volume 24(3) pages 397-417,1977

[9] J. Rocchio, Relevance feedback in information retrieval. The Smart Retrieval system—Experiments in Automatic Document Processing, Prentice Hall, 1971

[10] G. Salton, C. Buckley, Improving retrieval performance by relevance feedback, Journal of the American Society for Information Science volume 41(4)pages 288-297,1990

[11] C. Buckley, G. Salton, J. Allan, A. Singhal, Automatic query expansion using SMART, Proceedings of the 3rd Text REtrieval Conference, 1994

[12] A. Singhal, M. Mitra, C. Buckley, Improving Automatic Query Expansion, SIGIR'98, 1998

[13] A. Lu, M. Ayoub, J. Dong, Ad hoc experiments using EUREKA, Proceedings of the 6th Text Retrieval Conference,1997

[14] J. Xu, W.B. Croft, Query expansion using local and global document analysis, Proceedings of the 17th ACM SIGIR,1994

[15] J. Xu, W. B. Croft, Improving the effectiveness of information retrieval with local context analysis, ACM Transactions on Information Systems,2000.

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

[17] G., Salton, Automatic Text Processing – the Transformation, Analysis and Retrieval of Information by Computer. Addison –Wesley Publishing Co., Reading, MA, 1989.

[18] C., Buckley, G., Salton , and James Allan. Automatic retrieval with locality information using SMART. In D. K. Harman, editor, Proceedings of the First Text Retrieval conference (TREC-1), pages 59-72. NIST Special Publication 500-207, March 1993.

[19] C., Buckley, J., Allan, and G., Salton. Automatic routing and ad-hoc retrieval using SMART : TREC 2. In D. K. Harman, editor, Proceedings of the Second Text Retrieval conference (TREC-2), pages 45-56. NIST Special Publication 500-215, March 1994.

[20] G., Salton, Automatic Text Processing – the Transformation, Analysis and retrieval of Information by Computer. Addison –Wesley Publishing Co., Reading, MA, 1989.

[21] J., Rocchio, The SMART Retrieval System Experiments in Automatic Document Processing, Chapter: Relevance Feedback in Information Retrieval, 313{323, Prentice Hall, 1971.

Author's Information

A. A. Aly – Computer Science Department, El - Minia University; e-mail: abdelmgeid@yahoo.com