ELECTRONIC WORKSHOPS IN COMPUTING

Series edited by Professor C.J. van Rijsbergen

Jonathan Furner, School of Information and Media Studies, and David Harper, School of Computer and Mathematical Studies, The Robert Gordon University, Aberdeen, Scotland. (Eds)

Information Retrieval Research

Proceedings of the 19th Annual BCS-IRSG Colloquium on IR Research,
Aberdeen, Scotland, 8-9 April 1997

Paper:

Using Combination of Evidence for Term Expansion

R. Wilkinson

Published in collaboration with the
British Computer Society

Using Combination of Evidence for Term Expansion

Ross Wilkinson

Department of Computer Science, Royal Melbourne Institute of Technology (ross@cs.rmit.edu.au)
Melbourne, Australia

1997

Abstract

Expanding a user query automatically with terms taken from documents that are most similar to the query is a reliable way of finding more relevant documents. To date most approaches to this problem have focused on modifying the query. In this paper we argue that it is useful to create a new query from similar documents, rank both the user query and the new query, and combine the evidence. We show that there are both theoretic and practical advantages in this process.

Key words: Information Retrieval, Term Expansion, Combination of Evidence

1 Introduction

When we wish to assign a measure of similarity between two objects we may consider the similarity in many ways, the objects may be viewed as a whole in a number of ways, the objects may be decomposed, or the ways of measuring similarity may vary. Each of these methods can be considered as providing a piece of evidence with regard to relevance. The ways these different pieces of evidence might be used are varied also. We might add numbers, apply regression, apply complicated logical models, or develop new formulas.

Consider one example situation - term expansion. Term expansion occurs when a query is modified on the basis of the assumption that the most highly ranked documents contain terms that might be useful in retrieving other documents that the user would not locate on the basis of the initial query.

However, we may regard terms from a query, and terms from documents as two different sorts of evidence. In this paper we will explore both why a method of combination of evidence might be appropriate, and then consider some experimental evidence that suggest that this approach may be helpful.

We will first give a description of the method of term expansion, some of the variations that have been applied, and some of the problems associated with the technique. We then describe the technique of combination of evidence, along with a new justification of the approach that seems particularly relevant to term expansion. We then give a description of the experimental evidence that shows the approach to be useful and robust.

2 Term Expansion

In this paper text retrieval will be described in the context of the vector space model [12], a model in which all documents are represented as n -dimensional vectors where n is the size of the vocabulary. Each element of the vector is a non-negative real number, and sometimes the length of these vectors are required to be 1. This model allows queries and documents to be compared by measuring the cosine of the angle between them. Thus a query can be compared against all of the documents in a database, and the result is a ranked list of documents.

We use the cosine formula for measuring similarity:

$$\cos(q, d) = \frac{\sum_{t \in q \cap d} (w_{q,t} \cdot w_{d,t})}{\sqrt{\left(\sum_{t \in q} (w_{q,t})^2 \cdot \sum_{t \in d} (w_{d,t})^2\right)}}$$

with weights that have been shown to be robust and give good retrieval performance [2]: $w_{q,t} = \log(N/f_t) + 1$ and $w_{d,t} = \log(f_{d,t} + 1)$ where $f_{x,t}$ is the frequency of t in x , N is the number of documents in the collection, and f_t is the number of documents containing t .

Relevance feedback is an important technique in information retrieval that takes advantage of feedback from users as to whether they find documents presented to them as relevant or irrelevant. The technique is aimed at modifying the initial user query towards an optimum query that leads to ranking all relevant documents above all irrelevant documents. This optimum query is not in fact obtainable in the vector space model as the terms used to describe documents may not be expressive enough to separate relevant documents from irrelevant documents.

The standard method of modifying the query is to use the Rocchio formula [10]:

$$a * (\text{Original Query Vector}) + b * (\text{Average of Relevant Document Vectors}) - c * (\text{Average of Irrelevant Document Vectors})$$

Each of a , b , and c are non-negative real numbers. Typical values might be $a = 2$, $b = 1$ and $c = 0$. (The use of negative evidence is quite unreliable unless large samples are used.) This formulation does not distinguish whether terms come from query or documents other than by assigning different weights.

Most initial user queries are short. The terms in these queries may often have very little overlap with the terms in some of the relevant documents. Thus it is desirable to expand the set of terms used in a query. We

have just seen that this is one of the consequences of relevance feedback. However there may be no relevance information available from the user. Despite this there are successful term expansion techniques available. Most involve evaluating the initial query, assuming that the top N documents are relevant, and selecting terms from these N documents to augment the initial query.

The simplest method of doing this is to take the M most common terms from the N documents and add them to the initial query. In recent experiments in TREC [5] groups obtained better retrieval effectiveness in a variety of ways, although all can be related to Rocchio's method described above. The principal variations are in terms of the number of documents selected, the number of terms selected, and the weight that these terms are given. Lee [7] takes the view that no single approach is best and investigates combining the results of several different term expansions to good effect. Other methods of term expansion have been explored such as thesaural expansion but it is very difficult to obtain gains in these ways.

A recent survey of term expansion [3] showed that there had been a wide variety of approaches investigated. Amongst the approaches to term expansion, terms are selected from documents, thesauruses, and by users. The approaches investigate how to augment the initial query, or sometimes, how to replace it. The idea of developing multiple queries has not been explored, to the author's knowledge since early work on the SMART system [1].

Term expansion has consistently given improvement in retrieval effectiveness, however there are several problems that deserve attention. The first problem is that the standard technique of term expansion uses rather a large number of parameters. These will be detailed later, but any retrieval technique that requires setting several parameters, is exposed to the risk that the parameters that work on a test collection will not be appropriate to a working database of documents.

Another problem is that while both documents and queries may be written using English words, say, the nature of the usage may be quite different. The single occurrence of a word in a document may be quite peripheral to the central focus of the document, whereas this is less likely to be the case in a query. This problem may be ameliorated by using frequency counts, but it is nevertheless the case that the purpose of words in a query is not the same as in a document, in general. A word in a query appears in the context of the other words in the query, and similarly for words in a document. When combining these sets of words, some of the context is lost. We shall examine how combination of evidence may help address these problems.

3 Combination of Evidence

A simple model of the document retrieval process is that it involves simply an *indexer* and a *matcher*. Documents are passed to the indexer to obtain a set of representatives. Queries are similarly processed. The representatives of the documents and the queries are compared by the matcher and a result, usually a score is produced. For example the indexer may produce a vector of weights, representing a list of stopped and stemmed terms. The matcher may evaluate the cosine of the angle between two vectors. The scores produced by the matcher can be ordered, so that documents can be *ranked* and presented in order of their scores.

Figure 1: Simple Retrieval Model

However, there is no known method for ranking documents in exactly the order in which a user would want. Thus there has been much work on developing new indexers and new matchers that provide better ranking. The annual TREC experiment [5] has shown how researchers have been very successful in developing these strategies. Naturally different strategies have been developed, and some researchers have tried to use these different strategies together by *combining* the results of the strategies to produce a new ranking.

Figure 2: Combined Retrieval Model

There have been four reasons proposed for this approach. First, by allowing different strategies to be applied, it allows a more powerful query language to be used, such as is available with Inquiry [14]. This allows users to formulate queries in the widely different ways that they prefer if they are allowed [13]. Secondly, if documents are attempts to communicate, they inevitably have a component of noise. Each method of retrieval developed has the risk of attenuating some of the noise component. Thus it is possible that using several techniques and then combining, noise may be reduced. Similarly, rankings can be regarded as sources of evidence, and the more evidence of relevance the better. Hull et al. give a nice discussion of this [6]. Thirdly combination provides a convenient way of taking advantage of training. Different sources of evidence may be combined using regression, neural nets, or other methods derived from the machine learning literature [6][9]. Finally it may be that measures have quite different theoretic bases that are not easily comparable such as a cosine and a probability measure. In this case ranks can be combined as there is no obvious way of using the strengths of both approaches [15].

There are two other reasons that we believe makes combination of evidence particularly appropriate to term expansion, one to do with the nature of the evidence that is available, the other is due to the nature of relevance. Consider a query that is a structured document. In this case the document might have a title, an abstract and a set of paragraphs. We can form a query by simply merging the terms from all of the components. The consequence of this is that we lose the structure of the query, and in particular we treat terms that occur in different parts of the document/query in exactly the same way, even though they may well play a different role in the query. In the case of term expansion, again the terms come from quite different sources, the user issuing a query, and the writers of the documents. We may weight them differently depending upon the source, but we regard them as having the same role when we simply combine.

Documents are relevant to a query for a variety of reasons. Occasionally, a single document will provide a fact that satisfies a query. On other occasions different information will need to be combined from different documents. Thus, if documents are relevant to a query, they may well not be similar to each other. The

consequence of this is that any particular query can be close to only one of the peaks of relevance - the others are forced to be less similar. In the worst case if two documents are relevant to a query, and there is another document between them, such a document must score better than one of them using the cosine model. Only if all relevant documents are clustered together will the cosine model be able to provide optimum retrieval.

However, if one uses combination of evidence, one does not simply find an average vector, one can consider a set of cosines. The method of combination used can take into account of these peaks of relevance in ways that does not force only one peak of similarity, and hence relevance.

How do we combine evidence? There are many possibilities, but two methods have been used predominantly. First one can provide weighted averages of some form of normalized similarity. Second one can simply use the rank order of the documents. A recent study considers a range of these methods of combination [8]. In our study we used a weighted average of normalized similarity.

4 Experimental Design

In order to evaluate a hypothesis in information retrieval, we usually obtain a sample set of documents, a set of queries that can be posed against the documents, and a set of judgements by humans of the relevance or otherwise of documents to queries. A test is performed in which all of the documents are ranked against a query. This ranking is compared against the ideal ordering of all relevant documents being retrieved before all irrelevant documents. The ranking is evaluated using the tools of *recall*, the proportion of relevant documents retrieved, and *precision*, the proportion of retrieved documents that are relevant.

Because it is not always the case that all documents are evaluated as being relevant or irrelevant, two strategies are adopted. In one case all documents that are retrieved by a number of methods, down to some level, are evaluated. The remaining documents are assumed to be irrelevant. The other strategy is to only provide precision figures and ensure there are relevance judgements for the highly ranked documents only. In practice, comparing practical retrieval experiments with either of these methods almost always gives the same results. If a test of statistical significance, such as the Wilcoxon test, is applied, results are quite reliable. Standard texts such as [12] give a detailed coverage of retrieval evaluation.

A test collection was chosen from the Tipster Databases used for the TREC experiments [5], namely the second set of the Wall Street Journal articles that has 74,520 full text articles. There were a large number of queries that could be used for these experiments. However many queries have a large number of thesaural terms that had been carefully selected by trained queriers. These queries could be argued to have had manual term expansion already applied and so results would be less applicable compared to the more commonly practically observed phenomenon of very few query terms in an initial query. For this reason queries 101 to 200 were selected. These queries have an average of 76 query terms, after stop word removal and stemming. This number is still very high, but represented a legitimate experimental retrieval environment. The queries had several sections. We also experimented using just the title of the queries, and also just the description fields. The titles had an average of 4 terms after stopping and stemming. The descriptions had an average of 10 terms after stopping and stemming.

Standard methods of retrieval have been applied to this data, and have achieved good retrieval results [5]. Thus the experiment of using a standard retrieval method against this data represents a good baseline experiment against which various modifications can be tried.

The TREC queries are broken into three parts, sometimes with other fields as well. The first field is the title. It was not designed as a query of itself. The next field is a description of the information need. This is usually fairly terse, with perhaps 15 words in one or two sentences. There is then a narrative field which provides a detailed description of the information need. Our baseline experiments use all of this data.

The next set of experiments use just the descriptions - the closest approximation available of how a person might express their information need carefully and succinctly. While even these descriptions are longer than average queries, they do appear to represent quite realistic query descriptions. The narratives, on the other hand, are meant to represent what a user might say to a librarian in order for the librarian to issue a query.

5 Experiments

In our first experiment, the cosine measure given earlier was used to match the 100 queries against the 75,000 documents from the Wall Street Journal. Several other measures were applied, but none were superior. Thus this experiment gave a reasonable baseline to try to improve upon. Results are given as precision figures at 6 levels of recall, and an average.

0%	20%	40%	60%	80%	100%	Av.
0.724	0.544	0.400	0.250	0.123	0.039	0.323

Next we experimented to find a good set of expansion terms that could be added to the query. There are 4 different parameters that may be varied: the number of documents to be used, the number of terms to be selected for the expansion, the selection formula, and the comparative weight of the original query terms to the expansion terms. Unfortunately there is no obvious theoretical basis for determining these parameters, so past experience and much experimentation is needed.

Tests were carried out using between 10 and 50 documents, 10 and all terms, the formulas (*Freq. in top N docs*), (*Freq. in top N docs*)/(20 + *Freq. in all docs*), and (*Freq. in top N docs*)/log(1 + *Freq. in all docs*) were used, and comparative weights were varied by repeating the expansion terms 4 times, to repeating the original terms 4 times. Of these parameters, the selection formula was most important, and if terms were selected on the basis of the third formula listed above, a consistent gain was possible. The gains were only of the order of 1% to 8%. The best result was achieved by selecting the 40 best terms from the top 15 documents and doubling the occurrence of the original query terms.

0%	20%	40%	60%	80%	100%	Av.	Gain
0.743	0.564	0.424	0.276	0.153	0.045	0.349	8.0%

Naturally there is no guarantee that these parameters are appropriate to other collections and query sets. Moreover while good performance improvements are available some of the time, the improvement will not always be available. Further, the words in the query perform a different role to the words in the documents, so it was not clear that that should be simply merged. Thus we turned to methods of combination of evidence.

To combine we need to check that the new evidence is useful. Thus a run was carried out using only the expansion terms, without the original query.

0%	20%	40%	60%	80%	100%	Av.	Gain
0.746	0.517	0.344	0.154	0.047	0.012	0.279	-13.8%

Now while this run gives worse results than the original query it does rank different documents more highly, so that it is possible that combination of evidence may prove to be helpful.

In order to reduce the number of parameters, *all* terms in the top 15 documents were used as a query, again without the original query.

0%	20%	40%	60%	80%	100%	Av.	Gain
0.732	0.491	0.316	0.133	0.041	0.014	0.257	-18.7%

Now we are ready to combine. As has been seen, there are many ways of combining. The simplest is to use a weighted sum of normalized scores, $\alpha(S1) + (1 - \alpha)(S2)$. (Scores can be normalized by dividing each score by the maximum score for that query and matcher. Thus the top score will always be 1 and the other scores will be between 0 and 1.) α was varied between 0.5 and 0.95 and gains were consistently obtained. If the original query scores were combined with the scores using a query of the 100 best terms, using $\alpha = 0.8$, gave the best average precision, 0.342. However using all the terms in the top 15 documents gave consistently better results. Using $\alpha = 0.8$ we obtain:

0%	20%	40%	60%	80%	100%	Av.	Gain
0.758	0.587	0.448	0.268	0.125	0.038	0.352	12.6%

We thus have a significant improvement in precision and have just 2 parameters to select, the number of documents used for expansion, and α , the relative importance of the query and the documents. Both parameters are relatively stable for this collection, so that good improvement is available for a range of parameter settings. It may surprise that the bigger gains after combination of evidence occurred with the source of evidence that was, on its own, not as good as the other. The reason is that there is a bigger difference in the documents being identified by the source using all terms, so there is more scope for combination. In statistical terms, the smaller expansion is more correlated with the original query so provides less opportunity to improve.

The major draw-back however is a performance issue. Most retrieval systems have close to linear response time in the number of query terms. The queries being issued using all terms in the top 15 documents runs into the hundreds. Of course it is possible to do the expansion while the top document is being viewed – this is certainly enough time. However it is reasonable to sacrifice a little precision for speed, and thus select fewer terms. Thus for the remainder of the experiments we used 45 terms from the top 10 documents selected by *(Freq. in top 10 docs.)/log(1 + Freq. in all docs.)*

The remaining experiments were designed to test whether query length had any effect on the benefit of this approach to term expansion. Thus two new sets of queries were used: the titles of the topics only and the descriptions only. We ran a baseline experiment(BASE), then formed expansion sets, ran these on their own(EXP), then simply merged the two sets to form an expanded query(MERGE), and compared with the combination method described above(COMB).

Titles:

EXP.	0%	20%	40%	60%	80%	100%	Av.	Gain
BASE	0.566	0.324	0.225	0.115	0.049	0.013	0.186	-
EXP	0.579	0.361	0.240	0.110	0.029	0.009	0.195	4.8%
MERGE	0.603	0.398	0.286	0.166	0.069	0.015	0.230	23.8%
COMB	0.601	0.394	0.285	0.163	0.053	0.019	0.229	23.2%

Descriptions:

EXP.	0%	20%	40%	60%	80%	100%	Av.	Gain
BASE	0.570	0.354	0.230	0.115	0.032	0.013	0.194	-
EXP	0.571	0.355	0.231	0.117	0.033	0.013	0.195	0.7%
MERGE	0.568	0.347	0.237	0.133	0.055	0.010	0.202	4.1%
COMB	0.616	0.411	0.294	0.149	0.042	0.018	0.232	19.9%

For queries involving just the titles, merging the query terms with the expansion terms works just as well as combination. Performance improves substantially for the descriptions. Note how much expansion of any form helps queries that involve few words. The final thing to note is that α was set to favour the initial query. It would appear that this is not so appropriate to small queries, as the expansion sets give better results than the initial query. However, we were not interested in what were optimal values for α , just whether combination could work robustly.

6 Conclusions

In this paper, we have investigated methods of automatically expanding user queries, to take advantage of the vocabulary of documents in the database that have a good match. We have seen a variety of methods provide useful improvements.

We have introduced the technique of combination of evidence as an important strategy for use in this problem domain, and have given a new justification for the use of combination. We have seen that combination of evidence imposes less constraints on our notion of relevance than is the case with, in particular,

the vector space model. We also saw that it allows the combination of disparate evidence in a manner that does not have the disadvantage of unlike sources of evidence being treated exactly the same.

We have further seen that the strategy of combination of evidence is both robust and requires less tuning of parameters, than other techniques for term expansion. We have not compared combination with the Rocchio formula and its derivatives. There are two reasons for this. The first is that the system we were experimenting with does not support this form of feedback. The other is that most other retrieval systems do not support this either. Thus one is forced to simply introduce new terms into the query, or manipulate the results of the retrieval, as we have chosen to do.

Thus we believe that this paper provides evidence of the utility of term expansion in a very robust manner that can be adopted by any retrieval system that provides ranked output.

Acknowledgements

This work has been carried out while on sabbatical at Ubilab, the Information Technology laboratory of the Union Bank of Switzerland. I am very grateful for the facilities that have been provided. I am particularly grateful for the opportunity I have had to discuss this research with Hans-Peter Frei, Gabriele Sonnenberger and Tore Bratvold.

References

- [1] A. Borodin, L. Kerr, and F. Lewis. Query splitting in relevance feedback systems. In Salton [11].
- [2] C. Buckley, G. Salton, and J. Allan. The effect of adding relevance information in a relevance feedback environment. In W.B. Croft and C.J. van Rijsbergen, editors, *Proceedings of the 17th Annual International Conference on Research and Development in Information Retrieval*, pages 292–300, Dublin, Ireland, July 3–6 1994. Springer-Verlag.
- [3] E. N. Ethimiadis. Query expansion. In M. E. Williams, editor, *Annual Review of Information Science and Technology*, pages 121–187. American Society of Information Science, Silver Spring, Maryland, 1996.
- [4] H.-P. Frei, D. Harman, P. Schauble, and R. Wilkinson, editors. *Proceedings of the 19th Annual International Conference on Research and Development in Information Retrieval*, Zurich, Switzerland, August 18–22 1996. ACM.
- [5] D. Harman, editor. *Proceedings of the Fourth Text Retrieval Conference*, Gaithersburg, Maryland, 1995.
- [6] D. A. Hull, J. O. Pedersen, and H. Schutze. Method combination for document filtering. In Frei et al. [4], pages 279–287.
- [7] J. H. Lee. Combining multiple evidence from different properties of weighting schemes. In E. A. Fox, P. Ingwersen, and R. Fidel, editors, *Proceedings of the 18th Annual International Conference on Research and Development in Information Retrieval*, pages 180–188, Seattle, U.S.A., July 9–13 1995. ACM.
- [8] J. H. Lee. Combining multiple evidence from different relevance feedback methods. In R. Topor and K. Tanaka, editors, *International Symposium on Database Systems for Advanced Applications*, Melbourne, 1997. To appear.
- [9] D. D. Lewis, R. E. Schapire, J. P. Callan, and R. Papka. Training algorithms for linear text classifiers. In Frei et al. [4], pages 298–306.
- [10] J. J. Rocchio. Relevance feedback in information retrieval. In Salton [11], pages 243–264.
- [11] G. Salton, editor. *The SMART RETRIEVAL SYSTEM*. Prentice Hall, New Jersey, 1971.

- [12] G. Salton. *Automatic Text Processing*. Addison-Wesley, Reading, Massachusetts, 1989.
- [13] T. Saracevic and P. Kantor. A study of information seeking and retrieving III: Searchers, searches, and overlap. *Journal of the American Society for Information Science*, 39(3):197–216, 1996.
- [14] H. Turtle and W. B. Croft. Evaluation of an inference network-based retrieval model. *ACM Transactions on Office Information Systems*, 9(3):187–222, 1991.
- [15] R. Wilkinson, J. Zobel, and R. Sacks-Davis. Similarity measures for short queries. In Harman [5], pages 277–286.