Improving MEDLINE Document Retrieval Using Automatic Query Expansion

Sooyoung Yoo and Jinwook Choi

Dept of Biomedical Engineering, College of Medicine,
Seoul National University, 28 Yongon-Dong Chongro-Gu,
Seoul, Korea
{yoosoo0,jinchoi}@snu.ac.kr

Abstract. In this study, we performed a comprehensive evaluation of pseudo-relevance feedback technique for automatic query expansion using OHSUMED test collection. The well-known term sorting methods for the selection of expansion terms were tested in our experiments. We also proposed a new term reweighting method for further performance improvements. Through the multiple sets of test, we suggested that local context analysis was probably the most effective method of selecting good expansion terms from a set of MEDLINE documents given enough feedback documents. Both term sorting and term reweighting method might need to be carefully considered to achieve maximum performance improvements.

Keywords: Pseudo-relevance feedback, MEDLINE, Term sorting method.

1 Introduction

Automatic query expansion and relevance feedback techniques have been proposed to address the query-document mismatch problem. Relevance feedback (RF) expands terms from the user-identified relevant documents. Pseudo-relevance feedback (PRF) expands terms from the top documents initially retrieved. Although RF is useful for searchers, the overall performance of PRF is better in terms of search performance and searcher satisfaction [1]. In this paper, we focus on PRF technique for improving MEDLINE document retrieval.

Typically, PRF assumes that the initially retrieved top R documents are relevant. It extracts candidate expansion terms from the top R documents, sorts them using a term sorting (scoring) technique, and appends the top-ranked E terms to the initial query with modified weights. However, the performance of PRF can be affected by the quality of the initial retrieval result, such as the number of pseudo-relevant documents (R), the number of expansion terms (E), the term sorting method, and the term reweighting method applied [2-5]. The R and E parameters are usually chosen by experiments on a particular test collection. For the domain-specific test collection called OHSUMED where the documents are short references to medical literature, the performance of PRF therefore needs to be evaluated against various factors affecting the retrieval accuracy.

D.H.-L. Goh et al. (Eds.): ICADL 2007, LNCS 4822, pp. 241–249, 2007.

In this study, using the OHSUMED test collection, we perform a comprehensive experimental evaluation for various well-known term sorting methods and different term reweighing methods. For each term reweighting method, the characteristics among different term sorting algorithms will be discussed.

2 Methods

2.1 Test Collection

We used OHSUMED [6] as a test collection. The test collection is a subset of the MEDLINE database, which is a bibliographic database of important, peer-reviewed medical literature maintained by the National Library of Medicine (NLM). It contains 348,566 MEDLINE references from 1987 to 1991, and 106 topics (queries) generated by actual physicians in the course of patient care. About 75% of the references contain title and abstracts, while the remainder has only titles. Each reference also contains human-assigned subject headings from the Medical Subject Headings (MeSH). Each query contains a brief statement about a patient, followed by the information need. The queries are generally terse. The relevance is judged to be "definitely relevant", "possibly relevant", or "non-relevant". For our experiments we assume only "definitely relevant" are relevant. Therefore, only 101 queries which have definitely relevant documents are used for our evaluation. We use title, abstract and MeSH fields to represent each document and the information need field to represent each query.

2.2 Baseline Retrieval System

The baseline retrieval system was developed using SMART stopwords and Lovins' stemmer [7]. We simply used single terms as index terms. It had been shown that the best document-query weighting scheme was ann.atn for OHSUMED collection [6]. However, in our preliminary experiments, we found out that Okapi BM25 similarity measure [8] worked 9.4% significantly better than ann.atn in terms of precision at 10 documents (absolute precision at 10 documents was 0.2861 for Okapi BM25 and 0.2614 for ann.atn) although there was no significant difference for other evaluation measures (pared t-test, p=0.05).

Therefore, we chose Okapi BM25 weighting scheme as our unexpanded baseline retrieval model. In Okapi BM25 formula, the initial top-ranked documents are retrieved by computing a similarity measure between a query q and a document d as follows:

$$sim(q,d) = \sum_{t \in q \wedge d} w_{d,t} \cdot w_{q,t} \cdot \tag{1}$$

with $w_{d,t} = \frac{(k_1+1)\cdot f_{d,t}}{K + f_{d,t}}$ and $w_{q,t} = \frac{(k_3+1)\cdot f_{q,t}}{k_3 + f_{q,t}} \cdot \log \frac{N - f_t + 0.5}{f_t + 0.5}$

where t is a term of query q, f_t is the number of documents containing the term t across the document collection that contains N documents and $f_{d,t}$ is the frequency of

the term *t* in document *d*. *K* is $k_1((1-b) + b \times dl/avdl)$. k_1, *b*, and k_3 are parameters set to 1.2, 0.75, and 1,000 respectively. *dl* and *avdl* are respectively the document length and average document length measured in some suitable unit.

2.3 Selection of Expansion Terms

After we extracted all candidate expansion terms from the top R documents initially retrieved, we selected high-ranked E expansion terms to be added to the original query. In order to rank all candidate terms, we evaluated various term sorting methods in our preliminary experiments. From the experiments, we chose six competing methods with different properties (i.e. low term overlapping) to be evaluated further in this paper. Following term sorting algorithms were not considered in this paper: frequency [4], modified F4point-5 (F4MODIFIED) [9], the new term selection value based on significance measure [8], Doszkocs' variant of CHI-squared (CHI1) [5], r_lohi [10], and idf.

The six term sorting methods to be compared were Rocchio weight based on the Vector Space Model [5], Kullback-Leibler Divergence (KLD) based on the information theory [5], Robertson Selection Value (RSV) [5], CHI-squared (CHI2) [5], Expected Mutual Information Measure (EMIM) based on probabilistic distribution analysis [10], and Local Context Analysis (LCA) utilizing co-occurrence with all query terms [11]. In RSV, we did not ignore the probability that a nonrelevant document contain a candidate term *t* since the performance was better than the performance of ignoring it. We replaced the non-relevant documents statistics with the collection level statistics because we did not have any information about non-relevant documents.

After sorting all candidate terms including original query terms using one of the above methods, top-ranked E new terms (threshold score > 0) were finally selected for query expansion.

2.4 Traditional Term Reweighting Techniques

We evaluate two popular traditional term reweighting methods and our variants described in the next section.

For probabilistic feedback, we use the modified Robertson/Sparck-Jones weight [8]. It reweights expansion terms as follows:

$$\frac{1}{3} \times \log\left(\frac{(r_t + 0.5)/(R - r_t + 0.5)}{(f_t - r_t + 0.5)/(N - f_t - R + r_t + 0.5)} \right). \quad (2)$$

where r_t is the number of pseudo-relevant documents containing term *t* and the same definitions are used as in the above Okapi BM25 formula. The original query terms are reweighted by the original Okapi weight. Our preliminary experiments showed that 1/3 downgrading of its original Okapi weight for expansion terms was significantly better than using itself on the OHSUMED test collection.

For vector space feedback, we use standard Rocchio's formula. In original Rocchio formula, the new weight w'_{qt} of term *t* after query expansion is assigned as: (we assume a positive feedback)

$$w'_{q,t} = \alpha \cdot w_{q,t} + \frac{\beta}{R} \cdot \sum_{k=1}^{R} w_{k,t} \cdot \quad (3)$$

where $w_{q,t}$ is the weight of term t in the unexpanded query and $w_{k,t}$ is the weight of term t in a pseudo-relevant document k (in our retrieval system, that is, $w_{d,t}$ component of the Okapi BM25 formula). The α and β tuning constants are set to 1.

2.5 New Term Reweighting Techniques

Within Rocchio feedback formula, two variants of term reweighting were devised by extending the ideas of [12] for comparison. The main idea is to reflect the result of a term sorting algorithm on term reweighting process.

First, instead of using original Rocchio weight reflecting term importance within the pseudo-relevant documents, we utilized rank position of a term in the sorted term list for assigning the relevance weight as follows.

$$w'_{q,t} = \alpha \cdot w_{q,t} + \beta \cdot rank\_norm\_score_t \cdot \quad (4)$$

The $rank\_norm\_socre_t$ is evenly decreasing score according to the rank position of term t in the sorted term list. The $rank\_norm\_score_t$ of term t is calculated as 1 – ($rank_t$ - 1) / |term_list| where $rank_t$ is the rank position of term t in the sorted term list and |term_list| is the number of terms in the term list expanded. We call this approach "*rank_norm*".

Second, we intended to reflect the phenomenon that "ordinary" is shared among many, while "outstanding" is less frequent [13] on deciding the relevance weight of a term. Through multiple sets of preliminary test, we hypothesized that only the small number of high ranked terms would be enough more important in terms of relevance. Based on the hypothesis, the following formula was devised.

$$w'_{q,t} = \alpha \cdot w_{q,t} + \beta \cdot rank\_group\_score_t \cdot \quad (5)$$

For calculating $rank\_group\_score_t$ of term t, the sorted terms are firstly divided into k groups. A group of terms is then simply given $rank\_group\_score_t$ from k to 1. We assign a relatively small number of high scores, and a relatively large number of small scores using the term partitioning method used in the referenced paper [13]. We call this approach "*rank_group_kX*" where *X* is the number of groups of terms. In our preliminary experiments, small values of k performed better for OHSUMED test collection. We therefore used $k = 2$ in this paper, i.e. giving terms of the first group twice *rank_group_score* score than terms of the second group.

In this study, we also fixed $\alpha = \beta = 1$ for both *rank_norm* and *rank_group_kX* term reweighting methods.

3 Results

We retrieved the top-ranked 100 documents for 101 queries, and evaluated the performance using mean average precision (MAP). The unexpanded baseline MAP was 0.2163. We measured the performance of PRF for a wide range of R (1,2,3,4, and

5 to 50 by 5) and E (5 to 80 by 5) parameters. From our experiments, the performance was generally the best when E was between 10 and 15 in OHSUMED test collection. However, R parameter could not be fixed easily. Given a fixed number of expansion terms (E = 15), we therefore showed the performance improvements over the unexpanded baseline against different number of pseudo-relevant documents.

Fig. 1 to 4 display the MAP percentage change over the unexpanded baseline on various number of pseudo-relevance documents for different term sorting algorithms where probabilistic feedback, standard Rocchio's feedback, *rank_norm*, and *rank_group_k2* term reweighting were applied respectively.

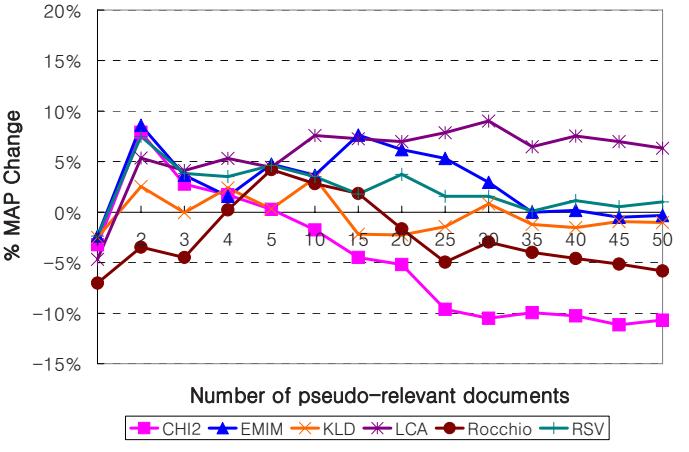

Fig. 1. Percent improvement in mean average precision with fixed E parameter (15 terms) for probabilistic term reweighting

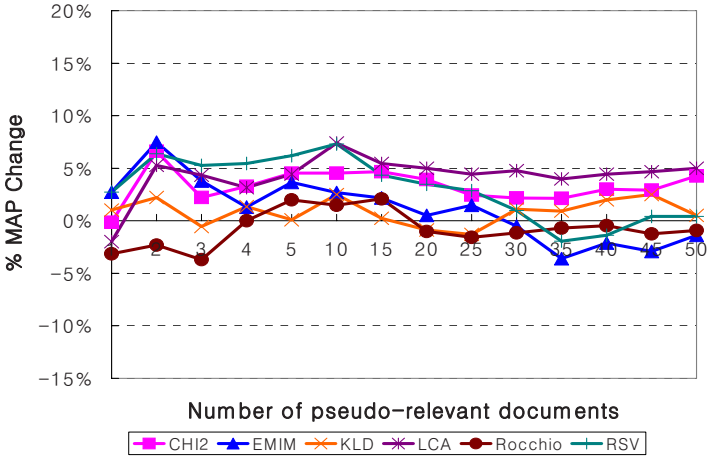

Fig. 2. Percent improvement in mean average precision with fixed E parameter (15 terms) for original Rocchio term reweighting

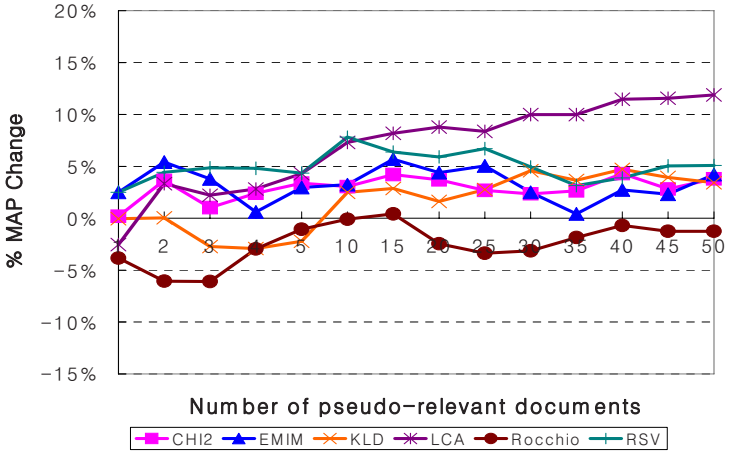

Fig. 3. Percent improvement in mean average precision with fixed E parameter (15 terms) for *rank_norm* term reweighting within Rocchio framework

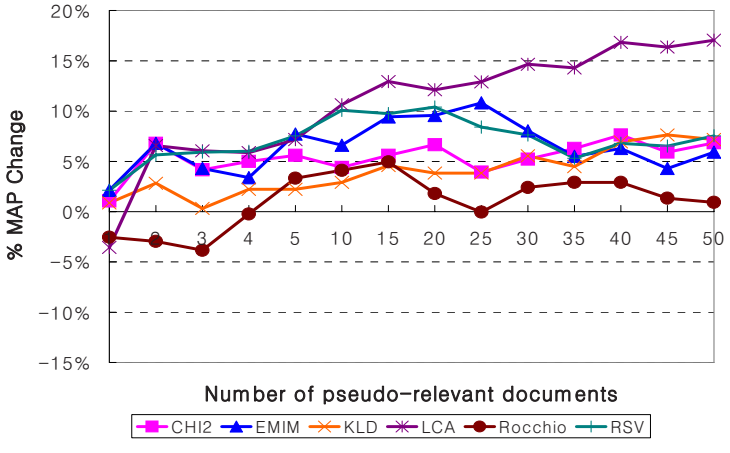

Fig. 4. Percent improvement in mean average precision with fixed E parameter (15 terms) for *rank_group_k2* term reweighting within Rocchio framework

In probabilistic term reweighting, the performance of competing term sorting algorithms was greatly affected by the R parameter settings as can be seen in Fig. 1. There was a noticeable decrease in the performance for CHI2 term sorting method when more than 25 documents were used for feedback on OHSUMED test collection. Overall, LCA term sorting method was less sensitive to R parameter settings with comparable or better performance than other term sorting strategies.

In standard Rocchio's term reweighting, well-known term sorting algorithms did not produce different performance patterns on a wide rage of R parameter as shown in Fig. 2. Although remarkable performance improvement could not be achieved for all term sorting algorithms, better performance improvement was expected for LCA and RSV.

In *rank_norm* and *rank_group_k2* term reweighting, the performance differences among different term sorting algorithms were distinguishable. As can be seen in Fig. 3 and 4, the performance of LCA was much better than the other methods on large R settings. On the other hand, Rocchio term sorting method showed worst performance. It seems that Rocchio method as a term sorting might not select good expansion terms from a set of MEDLINE documents.

It also can be seen that *Rank_group_k2* term reweighing method is better for the same term sorting method compared to *rank_norm* method. It supports our hypothesis that only top few terms of the sorted term list can be considered to be most important in determining their relevance weight. Therefore, it may be reasonable to divide terms into groups of more "good" terms and "less meaning" terms, rather than to differentiate their weight. It also seems that the difference of the relevance weights is less important.

Consequently, our experimental results suggest that LCA is probably the most effective method of selecting good expansion terms from a set of MEDLINE documents when feedback documents are given enough large. In addition, maximum performance improvements may be obtained by employing our *rank_gorup_k2* term reweighting rather than traditional feedback methods.

For the further analysis of individual queries, per-query improvements in MAP are given in Fig. 5. The differences in MAP between expanded query using LCA term sorting method and queries without expansion (baseline) are shown. Given fixed R = 50 and E = 15 parameters, each line is the performance differences for different term reweighting method. Our term rank-based reweighting scheme shows better performance than traditional probabilistic or Rocchio reweighting formula for more individual queries. It is proven that the reweighting methods affect the performance of individual queries and our reweighting methods are effective for more individual queries.

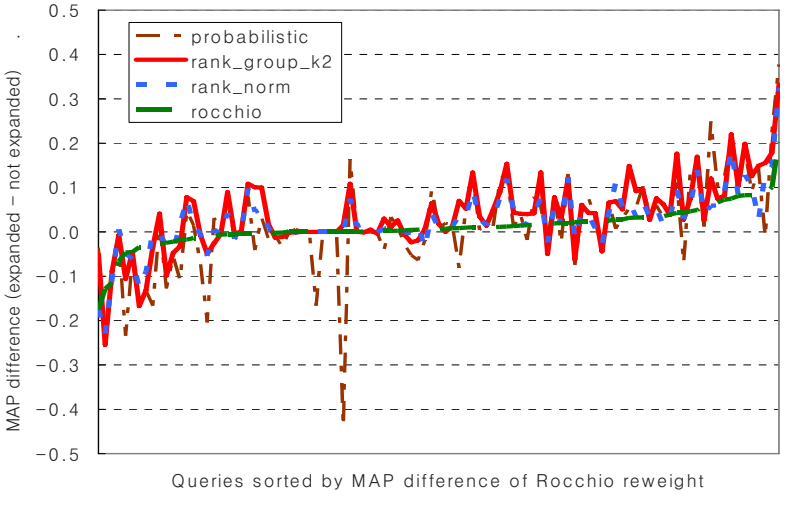

Fig. 5. Queries sorted by difference in mean average precision of original Rocchio term reweight for LCA term sorting method (R=50, E=15)

4 Discussion

For comparing well-known term sorting methods, LCA showed better performance than the other methods. It may be mainly due to the characteristics of OHSUMED queries itself. The queries frequently contain terms which represent a special medical task (e.g., “diagnosis”, “treatment”, “etiology”, etc). These terms are typically general. However, they can be effectively used for restricting query context. Since LCA considers co-occurrence with all query terms, it seems to implicitly restrict expansion terms to a specific medical task. Therefore, LCA will be suitable method for selecting expansion terms from a set of MEDLINE documents.

We tried to combine all pair-wise term sorting methods using standard combination methods [14] for further performance improvements. However, combing term sorting algorithms did not give any significant improvements over single best method in our experiments. More careful considerations may be needed when combining different methods in OHSUMED test collection.

5 Conclusion

In this paper, we performed a comprehensive experiment on PRF technique for a wide range of parameter choices using OHSUMED test collection. For the selection of expansion terms, LCA method utilizing co-occurrence with all query terms showed best performance when the pseudo-relevant documents were given large enough. Further performance improvements were achieved by applying our term rank-based reweighing variants within Rocchio framework rather than traditional probabilistic or original Rocchio formula. Therefore, both term sorting and term reweighting method might need to be carefully considered to achieve maximum performance improvements.

Acknowledgments. This work was supported in part by the Advanced Biomedical Research Center (ABRC) funded by KOSEF, and in part by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2006-(C1090-0602-0002)).

References

1. White, R.W.: Implicit feedback for interactive information retrieval. In: SIGIR Forum 2005, p. 70 (2005)
2. Fan, W., Luo, M., Wang, L., Xi, W., Fox, E.A.: Tuning before feedback: combining ranking discovery and blind feedback for robust retrieval. In: 27th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 138–145. ACM Press, New York (2004)
3. Jimmny, L., Murray, G.C.: Assessing the term independence assumption in blind relevance feedback. In: 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 635–636. ACM Press, New York (2005)

4. Harman, D.: Relevance feedback revisited. In: 15th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 1–10. ACM Press, New York (1992)
5. Carpineto, C., Mori, R., Romano, G., Bigi, B.: An Information-Theoretic Approach to Automatic Query Expansion. ACM Trans. Inf. Syst. 19, 1–27 (2001)
6. Hersh, W., Buckley, C., Leone, T.J., Hickam, D.: OHSUMED: an interactive retrieval evaluation and new large test collection for research. In: 17th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 192–201. Springer, Heidelberg (1994)
7. Lovins, J.B.: Development of a stemming algorithm. Mechanical Translation and Computational Linguistics. 11, 22–31 (1968)
8. Robertson, S.E., Walker, S.: Okapi/Keenbow at TREC-8. In: 8th Text REtrieval Conference (TREC-8), pp. 151–161 (1999)
9. Robertson, S.E.: On relevance weight estimation and query expansion. Journal of Documentation 42, 182–188 (1986)
10. Efthimiadis, E.N., Brion, P.V.: UCLA-Okapi at TREC-2: Query Expansion Experiments. In: 2nd Text REtrieval Conference (TREC.2), pp. 200–215, NIST Special Publication (1994)
11. Xu, J., Croft, W.B.: Improving the effectiveness of information retrieval with local context analysis. ACM Trans. Inf. Syst. 18, 79–112 (2000)
12. Carpineto, C., Romano, G.: Improving retrieval feedback with multiple term-ranking function combination. ACM Trans. Inf. Syst. 20, 259–290 (2002)
13. Anh, V.N., Moffat, A.: Simplified similarity scoring using term ranks. In: 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 226–233. ACM Press, New York (2005)
14. JH, L.: Analyses of multiple evidence combination. In: 20th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 267–276. ACM Press, New York (1997)