Query Expansion Using Web Access Log Files[1]

Yun Zhu and Le Gruenwald

The University of Oklahoma, 200 Falgar Street, Norman, OK 73019, USA
{zhujulie, ggruenwald}@ou.edu
http//www.cs.ou.edu/~database/faculty.htm

Abstract. Query Expansion has long been recognized as one of the effective methods in solving short queries and improving ranking accuracy in traditional IR research. Many variations of this method have been introduced throughout the past decades; however, few of them have incorporated web log information into the query expansion process. In this paper, we propose an expansion technique that expands document content at the initial index stage using queries extracted from the web log files. Our experimental results show that even with a minimal amount of real world log information available and a professionally cataloged knowledge structure to aid the search, there is still a significant improvement in using our query expansion method compared to the conventional query expansion ones.

1 Introduction

With the explosive growth of the World Wide Web, many web sites are providing web interfaces for users to access their databases. Thus, it is becoming increasingly important to find an optimum retrieval system suitable for such applications. Compared to the conventional full text retrieval systems, the web-interfaced retrieval systems face additional obstacles and opportunities that need to be addressed.

One of the well known problems posed to the web search systems is that web users tend to enter very short query terms, generally two to three keywords per query [11]. With the paucity of query terms, it is much more likely to have the word mismatch scenario where query terms do not match any keywords in a relevant document. Although these web-based search systems encounter the severe word mismatch problem, they also have an additional piece of information available that can be exploited for search purposes – web access log files recording user activities when they access the web site.

For decades, many studies in IR have tried to address the short query problem. One of the most recent and effective approaches is query expansion [5]. This technique involves expanding and reweighing query terms and reforming query results based on the expanded query. The source for term expansion is typically derived from user feedback, or documents assumed to be relevant to the original query or knowledge structure such as thesaurus. Although this technique has been

[1] This work was partially supported by the National Library of Medicine, Grant No. 1 G08 LM007877-01 and 1 G08 LM008054-01.

K.V. Andersen, J. Debenham, and R. Wagner (Eds.): DEXA 2005, LNCS 3588, pp. 686 – 695, 2005.

shown to be effective [5, 18], few studies have so far explored its application in web search and incorporation of web log information. The only experiment that utilizes log information was limited to expansion term selection [8]. However, there are more areas in the retrieval process where log information can be applied.

In this paper, we propose a new query association and expansion method that utilizes web log information to the full extent. By applying log mining techniques, we are able to expand the document contents with query terms entered by the users. We then perform expansion search based on the expanded document contents. Our query expansion method achieves good improvement compared to conventional expansion techniques even when there is professionally cataloged knowledge structure to aid the search and only a minimal amount of log information available.

2 Related Works

The concept of query expansion was originated in the 1970's [12] where user feedback is applied not only to the reweighing of terms, but also to the expansion of search terms for further retrieval improvements. The basic idea of this technique is that each time the system retrieves a set of documents based on a user's query, a set of extra terms is then selected from the relevant documents identified by the user, then finally a new query is formed with the selected terms and a new search is performed. This process can be carried out iteratively, and it is expected that the more iterations it goes through, the more number of relevant documents will be retrieved. Although this technique has been shown to be effective [13, 16], the requirement for users' constant feedback on relevance is not appealing.

Throughout the years, many variations of query expansion techniques have been introduced. They can be categorized into three main groups: manual query expansion [4], query expansion based on the complete document collection [5, 6, 7, 9] and query expansion based on local analysis [2, 17, 18] (also referred to as *Relevance Feedback* or *Pseudo Relevance Feedback*). The manual query expansion involves users' judgment on which terms to select for expansion. This technique is rarely implemented because studies have shown that it does not improve search results effectively [4]. The basic idea for query expansion based on the complete document collection is to study the correlation between terms in the documents and identify the relationships between terms throughout the collection. This technique usually involves the manual or automatic building of a thesaurus type knowledge structure to aid the search. Unfortunately, building such a knowledge structure is extremely expensive, and has not achieved any significant improvements in experiments [5]. The third group, query expansion based on local analysis, assumes that the top n documents retrieved based on users' original queries are always relevant, thus the expansion terms are selected via studying the correlations between the terms within those n documents. This technique dismisses users from any form of input, and has shown improvements in many studies, especially in the experiments from TREC [5, 18].

Recently, there have been two interesting studies on applying alternative pseudo relevance feedback techniques to web search and both have achieved significant improvement. The first study by Billerbeck and others [3] utilizes a query association

technique, which relates a historical query term with the top *n* ranked documents retrieved using the Okapi ranking formula [13,18]. At the end of the process, each document is represented by a surrogate file formed by a set of query terms associated with it. The surrogate files then become the base for the initial document retrieval and term selection. The retrieval based on the expanded query is later performed on the original document collection. The basic idea is that queries are usually well thought out descriptions of what the users are looking for, so the surrogate files formed by those queries are well defined abstracts of the document, and thus improving the final search results. The researchers claimed an improvement of 18-20% over the conventional expansion approach. Although this technique has achieved promising results, there are several issues that need further improvement. First, the initial formation of the surrogate files uses a ranking formula to identify the association, thus the reliability of the surrogate files depends on how perfect the ranking formula is, and it does not reflect the true desire of the users. Second, this technique requires a complete query set that is appropriate for the document collection; otherwise, the quality of the surrogate files will not be good enough to improve the search.

The second study aims at improving query expansion using web log information [8]. The researchers developed a probabilistic term selection and reweighing function that determines expansion terms not only on their original weight, but also on the probability of both the terms and the document appearing together in a user session. The motivation behind this technique is that the researchers believed the correlations between the query terms and documents extracted from a web log are valuable user feedback and more reliable than the pseudo relevance feedback. The experiment results showed a substantial improvement over the pseudo relevance feedback approach. However, examining their formula closely, we have found that if a term does not appear in the content of a document, then no matter how relevant that term is to the document, it will always be assigned a zero probability for that document. That is to say, the formula does not capture the terms that are deemed relevant by the users, but do not appear in the document contents. Furthermore, this technique also requires a web log that has sufficient document coverage and content coverage from the queries; otherwise, the system will not be able to obtain any term for expansion.

3 Log-Based Association Expansion

The Log-based Association Expansion approach we propose in this paper is aimed at eliminating the limitations of the two approaches discussed in Section 2. In our approach, we extract associations between users' queries and documents from the web log, which is a more reliable source of user feedback compared to query association. We then make full use of these extracted associations by not only applying them to expansion term selection, but also to original document content expansion. The process can be broken down into two phases: document expansion by web log mining and query expansion retrieval. Their detailed descriptions are as follows:

Phase 1. Document Expansion by Web Log Mining – The system expands document contents with associated query terms extracted from the web log using a mining algorithm.

In this phase, the system first parses the web log and extracts the information such as IP addresses, timestamps, document IDs and search keywords entered. The IP addresses and timestamps are then used to identify user sessions. This paper uses a 30 minutes timeout to identify the sessions – an approach that has been considered in the literature as a default approach for session identification [19].

The system then runs a mining algorithm on the extracted information to identify the associations between query terms and documents. We implemented a mining algorithm (shown in Figure 1) inspired by the Apriori data mining technique [1] which uses a user-defined minimum support threshold (min_sup) to identify frequent item sets (a set of items the frequency of each is equal to or higher than min_sup). Here frequency refers to the number of sessions in which a query-document pair appears together. The first part of the algorithm (lines 1 to 5) is aimed at identifying frequent query-document pairs. It first obtains the frequency for each pair, and if the frequency is higher than or equal to min_sup (line 4), every term in the query is added to the document with the corresponding frequency. The algorithm then handles non-frequent query-document pairs (lines 6 to 9): if the query term exists in the document, it is also added to the document. The assumption is that the single threshold value might have left out relevant query terms that have not yet accumulated enough frequency, thus by adding terms that exist in the documents, we partially salvage relevant terms with low frequencies without risking adding irrelevant terms.

Fig. 1. Mining algorithm for the association identification

```
1 For each document d_i and query q_i
2    Count frequency n as the number of sessions where d_i and q_i
3    appear together
4    If (n/total_session_count>=min_sup)
5      Add every term x_j in q_i to d_i for n times
6    Else
7       For each x_j in q_i that was not added to the document
8         If (x_j appears in d_i )
9            Add x_j to d_i for 1 time
```

At the end of this web log mining process, we obtain a set of documents expanded by users' query terms. There are several advantages for this addition. First, the query terms and associations identified by the process are a true reflection of user's judgment on relevance feedback and, therefore, are more reliable than the associations created by a ranking algorithm. Second, the process adds terms to the original document rather than building a separate file out of query terms. This not only enables the early use of log information in the web development stage, but also ensures the full coverage of document contents where there is none or incomplete log information available for the document. Third, by adding query terms within the frequent item set to the document, we are able to capture terms that might not exist in the document,

yet relevant to the document contents. The web log mining process should be repeated periodically when a sufficient amount of log information has been accumulated.

Phase 2. Query Expansion Retrieval – When a user's query Q arrives, the system performs query expansion search based on the expanded document derived from Phase 1. The query expansion search process is carried out in the following steps:

a. Initial Retrieval Based on Q – The system uses Formula (1) to measure each document's score *W* with respect to *Q*, and retrieves a set of documents ordered by their scores from the highest to the lowest. This formula is a modified version of the Okapi formula [13], where the values of the parameters are with respect to the expanded document rather than the original document. For each document, the formula sums the weight (w_i) for each term that appear in both the document and the query *Q* and adjust the weight with a normalizing factor. The documents are then ranked according to the result in descending order. It is worth noting that both w_i and *W* are inherently influenced by the query terms added in the expanded document. Basically, terms that reflect users' desire is more likely to have a higher weight in the document, and documents containing those terms are more likely to obtain higher scores when there is a match between the terms in the document and the query terms.

$$W = \sum_{i \in d \cap q} w_i \times \frac{TF(k_1 + 1)}{K + TF} \tag{1}$$

where *d* is the set of terms in the expanded document, *q* is the set of terms in the query Q, *TF* is the number of times the term appears in the expanded document, $w_i = \log \frac{(N - n + 0.5)}{(n + 0.5)}$ when relevant information (documents known to be relevant) is not available, $w_i = \log \frac{(r + 0.5)(N - n - R + r + 0.5)}{(R - r + 0.5)(n - r + 0.5)}$ when relevant information is available, R is the number of relevant expanded documents, r is the number of relevant expanded documents containing the term, *N* is the number of expanded documents, *n* is the number of expanded documents that contain the term, $K = k_1((1-b) + b\frac{DL}{AVDL})$, *DL* is the length of the expanded document, *AVDL* is the average length of the expanded documents in the given collection, and *b*, k_1 are tuning constants usually set to 0.75 and 1.2, respectively.

b. Expansion Term Selection – The system selects expansion query terms from the set of terms within the top *n* documents (denoted as *T*) derived from step a. During this step, each term *i* within *T* is weighted using Formula (2).The terms with the highest weights and do not belong to the original query are selected to form a new query.

$$\ln(\sum_{i \in T} w_i + 1) \tag{2}$$

c. Expanded Search and Result Sets Merging – Using Formula (1), the system performs another search based on the new query derived from step b. Then it combines the retrieved document set with the original document set derived from step a. In order not to overpower the new query results with the original query, we use formula (3) to obtain the weight for each of the document in the final result set:

$$W_{orig} + \alpha \cdot W_{new} \tag{3}$$

where W_{orig} is the document weight based on the original query, W_{new} is the document weight based on the new query, and α is a value from 0 to 1 to adjust the influence of the new weight on the result. The final result set is displayed in the decreasing order of document weights.

Our technique differs from Billerbeck's [3] in that we expand each document with queries extracted from the web log instead of building a new surrogate file based on associations. We believe our technique uses a more reliable source for query expansion because web logs give a true reflection of users' judgments. Comparing with Cui's [8], which also utilizes web logs, our technique expands the usage of log information in that it is not only applied to expansion term selection, but also to document contents expansion at the initial phase. Thus web log information influences every step of the expansion process starting from the first retrieval based on the user's original query. Furthermore, our web mining technique also enables the addition of new and related terms to the document contents. Another advantage of our technique is that it can be implemented even when there is not enough log information available, and still fine-tunes the search performance, while the other approaches would not be able to produce any query expansion terms.

4 Experiments

4.1 Test Data Collection

Our document collection comes from a real world web site – HEAL (Health Education Assets Library) [10]. The web site hosts over 36,000 multimedia files for medical professionals. These multimedia files are organized using metadata tables that textually describe every aspect of the files, and web user interfaces are provided for users to search these files through the metadata tables. The document length for each description ranges from 2 words to 314 words. In order to improve the search performance, special medical catalogers assign each file to the MeSH (Medical Subject Headings) tree [12] and additional keywords table. All the techniques implemented in the experiments are aided by the MeSH tree and keyword tables.

The log access files were collected from the web site server from May 2002 to June 2004. Due to the fact that this web site is relatively new, the number of visits is incomparable to other popular web search sites. Among these log files, we were only able to identify 4,747 sessions that contain search, clicking or downloading activities. Compared to over four million sessions in Cui's experiments [8], our log information is extremely limited.

We used 50 queries to conduct the experiments. These 50 queries were selected by the medical professionals and researchers who have used the HEAL system out of 100 queries randomly generated from the log activities. The average length of those queries is 1.6 words. The relevance judgment was conducted by the medical professionals on a 2,000 experiment document collection, which was also randomly selected from the 36,000 files in the database. These 2,000 files form the test

collection for retrieval effectiveness testing. The term weights are assigned based on the complete database collection rather than the test collection.

4.2 Experiment Setup

All the formulas used in the experiments have been listed in Section 3. During these experiments the following retrieval techniques and test runs are studied on the same test collection for comparison purposes:

1. FreqCase – This search system uses Formula (1) to perform document retrieval and no expansion process or relevant information is involved.
2. LogCase – This is a variant of our proposed Log-based Association Expansion search system. It searches on the expanded documents using the algorithm from FreqCase.
3. ExpCase – This search system is the conventional query expansion search. In this paper, this approach has the same process as our Log-based Association Expansion, except that no log information was added.
4. ExpLogCase – This is our proposed Log-based Association Expansion search system described in Section 3. The system utilizes validated search terms and expansion search process to perform the search.
5. CuiCase – This is the test run for Cui's probabilistic query expansion using query logs [8].

Due to the limited amount of real world log activities, we have generated simulated log activities to study the effect of abundant and correct log activities on the performance of the above search systems. The simulated log activities were created via randomly selecting relevance feedback from the medical professionals' relevance judgments, and inserting them for a random number of times (ranging from 2-7 times) into the extracted log information. The following are the test runs using the simulated log information:

6. LogCaseSim – This is a test run for LogCase with simulated web logs.
7. ExpLogCaseSim – This is a test run for ExpLogCase with simulated web logs.
8. CuisCaseSim – This is a test run for CuiCase with simulated web logs.

Apart from the above test runs, we also tried test runs for Billerbeck's query expansion using query association [3]. Unfortunately, the association creation process was not successful due to the fact that there are insufficient historical queries to form proper surrogate files for each document. For this experiment, we have also tested exhaustively on dynamic parameter value settings to identify the ones that yield the best combination of search accuracy and response time. These values are 0.000632, 6, 2 and 1/3 for min_sup, number of top ranked documents for expansion, number of terms selected for expansion and α, respectively.

5 Experiment Results

For the experiments, we used the precision for the first 10, 20, 30, 40 and 50 documents retrieved (denoted in Table 2 as p@10...p@50), the average precision for

the first 50 files retrieved (denoted as Avg. P) the recall for the first 300 documents retrieved (denoted as Recall 300), and Average 11-point interpolated Recall-Precision (denoted as avg. 11-RP) as the evaluation metrics [2].

Table 2. P@10, 20, 30, 40, 50, Average Precision, Average 11-point precision, and recall 300

	p@10	p@20	p@30	p@40	p@50	Avg. P	Vs. FreqCase	Avg. 11-RP	Recall (300)
FreqCase	0.6220	0.5460	0.4713	0.4055	0.3560	0.4802	-	0.3299	0.4364
LogFreqCase	0.6900	0.6300	0.5687	0.5070	0.4560	0.5703	0.1878	0.4426	0.6065
ExpCase	0.7020	0.6370	0.5780	0.5160	0.4680	0.5802	0.2083	0.4804	0.6875
ExpLogCase	0.7200	0.6560	0.5987	0.5455	0.4940	0.6028	0.2555	0.5161	0.7381
CuiCase	0.6360	0.5770	0.5120	0.4605	0.4212	0.5213	0.0857	0.3922	0.5882
logFreqCaseSim	0.8580	0.7830	0.7120	0.6470	0.5972	0.7194	0.4983	0.6314	0.7780
ExpLogCaseSim	0.8480	0.7710	0.7100	0.6450	0.5928	0.7134	0.4857	0.6688	0.8615
CuiCaseSim	0.6380	0.6110	0.5820	0.5360	0.5008	0.5736	0.1945	0.5048	0.7769

From Table 2, we can see that for the first 50 files retrieved, all of the non-simulation runs gained significant improvement over the FreqCase in terms of Avg. P. Among them, LogFreqCase outperformed FreqCase by 18%. This shows that user log information can greatly improve the retrieval performance. Furthermore, such improvement can be further enhanced when query expansion is applied. Our technique, ExpLogCase, had the highest precision value for all the first N files retrieved, and gained 25% and 5.7% over FreqCase and LogFreqCase.

The Avg.P value for CuiCase is the lowest among all the non-simulation runs. It underperformed by 15% compared to ExpLogCase. There are several factors contributing to such low performance. First, the term selection in Cui's approach relies heavily on the web query logs, and the limited amount of logs available in this experiment affects the quality of expansion terms. Second, as mentioned earlier, Cui's approach missed out query terms that are relevant to the document but did not appear in the document. Our technique captures such terms, thus resulting in a better performance. Third, for this particular test data set, the vector space ranking algorithm in CuiCase underperforms by 5.5% in terms of Avg.P compared to the probabilistic model implemented in our technique. All the simulated runs experienced 20% to nearly 50% improvements in terms of Avg.P (See Table2) over the FreqCase. However, CuiCaseSim's Avg.P value was 24% lower compared to LogFreqCaseSim. This shows that our technique outperforms Cui's even when there is sufficient log information.

We also analyzed the test runs using the average 11-interpolated precision (Table 3). The test run for the proposed technique, ExpLogCase, again outperformed all the others. It had a 56%, 31%, 16.6% and 7.4% improvement over FreqCase, CuiCase, LogFreqCase and ExpCase, respectively. Furthermore, its simulated test run – ExpLogCaseSim outperformed LogFreqCaseSim and CuiCaseSim by 5.9% and 32%.

Table 3. Comparison Analysis for Average 11-point precision

	Vs. FreqCase	Vs. LogFreqCase	Vs. Expcase	Vs. LogFreqCaseSim	Vs. CuiCase	Vs. CuiCaseSim
FreqCase	-	-	-	-	-	-
LogFreqCase	0.3416	-	-	-	0.1286	-
ExpCase	0.4561	0.0853	-	-	0.2249	-
ExpLogCase	0.5643	0.1660	0.0743	-	0.3159	-
CuiCase	0.1888	-0.1139	-	-	-	-
logFreqCaseSim	-	-	-	-	-	0.2507
ExpLogCaseSim	-	-	-	0.0592	-	0.3247
CuiCaseSim	-	-	-	-	-	-

These figures show that overall, the expansion techniques outperform the non-expansion ones. Our technique, ExpLogCase, outperforms all the other techniques in terms of precision for the first 50 files retrieved, recall 300 and average 11-point precision even in situations where there is a very limited amount of extracted query associations available.

6 Conclusions

Various expansion retrieval techniques have explored retrieval effectiveness using original documents and rely on historical queries or surrogate files for further expansion. We have proposed a technique that incorporates the use of web log activities in improving information retrieval from the initial index term weighing stage. Our technique is robust enough to handle situations when not enough log activities are available. Our experiment has shown that the Log-base Association Expansion approach can achieve significant improvements using log information and outperforms similar approaches. Thus we can conclude that using log activities at the initial stage of expansion search does improve web search.

In future, we plan to further investigate the effect of the amount of log activities on the readjustment of various parameters in our technique. We also plan to study the efficiency and tradeoff between the initial log mining process and its impact on term weight readjustment and how often the term scores should be readjusted.

References

1. Agrawal, R., Imielinski, T., Swami, A.:Mining Association Rules Between Sets of Items in Large Databases. ACM SIGMOD, (1993) 1 – 10
2. Baeza-Yates, R., Ribeiro-Neto, B.:Modern Information Retrieval (Chapter 3).Addison Wesley, ACM Press, NY, (1999)
3. Billerbeck B., Scholer F. Williams H.E., Zobel, J.:Query Expansion using Associated Queries. CIKM, (2003) 2 – 9

4. Brajnik, G. Mizzaro, S., Tasso, C.: Evaluating User Interfaces to Information Retrieval Systems: A Case Study on User Support. ACM SIGIR, (1996) 128 – 136
5. Carpineto, C. Mori, R.D., Romano, G., Bigi, B.: An Information-Theoretic Approach to Automatic Query Expansion. ACM Transaction on Information Systems, 19(1) (2001) 1 - 27
6. Carpineto, C., Romano, G.: Effective Reformulation of Boolean Queries with Concept Lattices. FQAS 98, Roskilde, Demark, (1998) 83 – 94
7. Cooper, J.W., Byrd., R.J.: Lexical Navigation: Visually Prompted Query Expansion and Refinement. In Proceedings of the 2nd ACM International Conference on digital Libraries, (1997) 237 – 246
8. Cui, H., Wen, J.R., Nie, J.Y., Ma, W.Y.: Probablistic query expansion using query logs. The Eleventh International World Wide Web Conference, ACM, May, (2002) 325 – 332
9. Grefenstette, G.: Explorations in Automatic Thesaurus Discover. Kluwer Academic Publisher, MA, (1994)
10. HEAL: http://www.healcentral.org, accessed on June 7th, 2005
11. Jensen, B.J., Sprink, A., Scaracevic, T.: Real life, real users and real needs: A study and analysis of users' queries on the Web. Information Processing and Management. 36(2) (2000) 207 – 277
12. MeSH: http://www.nlm.nih.gov/mesh/meshhome.html, accessed on June 7th, 2005
13. Robertson, S.E., Walker, S.: Okapi/Keenbow at TREC-8. TREC-8, (1999) 151 – 162
14. Salton, G.: The SMART retrieval system (Chapter 14). Prentice- Hall, Englewood Cliffs, NJ., (1971)
15. Sparck Jones, K.: Experiments in relevance weighting of search terms. Information Processing and Management. 15 (1979) 133 – 144
16. Sparck Jones, K.: Search term relevance weighing given little relevance information. Information Processing and Management. 35 (1979) 30 – 48
17. Sparck Jones K., Walker S., Robertson, S.E.: A Probabilistic Model of Information Retrieval: Development and Comparative Experiments Part 1. Information Processing and Management, 36 (2000) 779 – 808
18. Sparck Jones K., Walker S., Robertson, S.E.: A Probabilistic Model of Information Retrieval: Development and Comparative Experiments Part 2. Information Processing and Management, 36 (2000) 809 – 840
19. Srivastava, J., Cooley, R., Deshpande, M., Tan, P.N.: Web Usage Mining, Discovery and Applications of Usage Patterns from the Web Data. SIGKDD Explorations, 1(2), (2000) 12 – 23