Probabilistic Query Expansion using Query Logs

Hang Cui*
Institute of System Engineering
Tianjin University
Tianjin, P.R.China
hang_cui@hotmail.com

Ji-Rong Wen
Microsoft Research Asia
3F, Beijing Sigma Center
No.49, Zhichun Road Haidian District
Beijing, P.R.China
jrwen@microsoft.com

Wei-Ying Ma
Microsoft Research Asia
3F, Beijing Sigma Center
No.49, Zhichun Road Haidian District
Beijing, P.R.China
wyma@microsoft.com

ABSTRACT

Query expansion has long been suggested as an effective way to resolve the short query and word mismatching problems. A number of query expansion methods have been proposed in the traditional information retrieval field. But these previous methods seldom take into account the characteristics of web searching. Especially, they do not take advantages of the user interaction information recorded in the web query logs. In this study, we put forward a new probabilistic method for query expansion based on query logs. The central idea of this method is to construct the probabilistic correlations between query terms and document terms through mining the query logs. Then the probabilistic correlations can be used to select high-quality expansion terms for the newly coming queries. The experimental results show that our log based probabilistic query expansion method can improve the search performance greatly and has overwhelming advantages over other existing methods. We also give the quantitative measure of the difference between query terms and document terms, which provides the experimental evidence to support the assumption that there is a big gap between the query space and the document space.

Categories and Subject Descriptors

H.3.3 Information Search and Retrieval: *Query Expansion, Log Mining*

General Terms

Algorithms, Management, Measurement, Performance, Design, Experimentation, Human Factors, Theory

Keywords

Query expansion, log mining, probabilistic model, information retrieval, search engine

1. INTRODUCTION

With the explosive growth of information on the World Wide Web, there is an acute need for search engine technology to help users to exploit such an extremely valuable resource. Despite the fact that keywords are not always good descriptors of contents, most existing search engines still rely solely on the keywords contained in the queries to search and rank relevant documents. This is one of the key factors that affect the precision of the search engines. In many cases, the answers returned by search engines are not relevant to the user's information need, although they do contain the same keywords as the query.

The web is not a well-organized information source and innumerous "authors" created and are creating their websites independently. Therefore, the "vocabularies" of the authors vary greatly. But users usually tend not to use the same terms appearing in the documents as search terms. This is a fundamental problem called term mismatch in information retrieval. Moreover, most words in natural language have inherent ambiguity. These reasons make accurate query formulation a rather difficult task for the web users.

It is also generally accepted that web users typically submit very short queries to the search engines and the average length of web queries is less than two words [16]. Therefore, such short queries usually lack sufficient words to cover useful search terms and thus heavily decrease the performance of web search in terms of both precision and recall.

To overcome the above problems, researchers have focused on using query expansion techniques to help users formulate what information is really needed. Query expansion involves adding new words and phrases to the existing search terms to generate an expanded query. Existing state-of-the-art query expansion approaches can mainly be classified 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. Global analysis requires corpus wide statistics such as the co-occurrences of every pair of terms, which results in a similarity matrix among terms. To expand a query, terms which are most similar to the query terms are identified and added. The global analysis techniques are relatively robust in average performance. But it requires corpus wide statistical data which consumes a considerable amount of computer resources. Moreover, since it only focuses on the document part and does not take into account the query part, global analysis cannot address the term mismatch problem well.

WWW 2002, May 7-11, 2002, Honolulu, Hawaii, USA.

* This work was performed while the author was a visiting student at Microsoft Research Asia.

Different from global analysis, local analysis uses only some initially retrieved documents for further query expansion. A more recent and well-known local analysis technique is relevance feedback [14], which modifies a query based on the relevance judgments of the retrieved documents, that is, only extract expansion terms from the relevant documents judged by the user. Relevant feedback can achieve very good performance only if the users afford sufficient and correct relevance judgments. Unfortunately, in a real search engine environment, 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 of lacking of sufficient relevance judgment information, pseudo relevance feedback (also known as blind feedback) is commonly used. The basic idea of pseudo feedback is to assume the top-ranked documents are relevant and then extract expansion terms from those documents. Thus actual input from the user is not required. Recent TREC results show that local feedback approaches are effective and, in some cases, outperform global analysis techniques [17]. Nevertheless, this method has an obvious drawback: if a large fraction of the top-ranked documents is actually non- relevant, then the words added to the query (drawn mostly from these documents) are likely to be unrelated to the topic and the quality of the documents retrieved using the expanded query is likely to be poor. Thus the effects of pseudo feedback depend heavily on the quality of initial retrieval.

Recently, Xu and Croft [18] proposed a local context analysis method, which aims to combine the local analysis and global analysis. First, noun groups are used as concepts which are selected based on co-occurrences with query terms. Then concepts are chosen from the top-ranked documents, similar to local feedback. Since expansion terms used here are not based on frequencies in the top-ranked documents but based on co-occurrence with terms in the query, the local context analysis method can overcome the difficulty of local analysis to some extent. To our knowledge, local context analysis is one of the expansion methods achieving the best performance so far [18]. However, local context analysis is based on the hypothesis that a common term from the top-ranked relevant documents will tend to co-occur with all query terms within the top-ranked documents. This is a reasonable but not always holding hypothesis.

In this study, we put forward a new query expansion method based on query logs. Through daily usages, every search engine website accumulates a large amount of query logs. From the query logs we can extract many query sessions. A query session is defined as follows:

session := <query text> [clicked document]*

Each session corresponds to one query and the documents the user clicked on. The central idea of our method is that if a set of documents is often selected for the same queries, then the terms in these documents are related to the terms of the queries. Thus some probabilistic correlations between query terms and document terms can be established based on the query logs. Then these probabilistic correlations can be used to select the high-quality expansion terms from the document space for the newly coming queries.

One important assumption behind this method is that the clicked documents are "relevant" to the query. Although the clicking information is not as accurate as explicit relevance judgment in traditional IR, the user's choice does suggest a certain degree of "relevance" of that document to his or her information need. In fact, users usually do not make the choice randomly. Even if some of the document clicks are erroneous, we can expect that most users do click on relevant documents. Our previous work on using query logs to cluster similar queries also strongly supports this assumption [16]. Therefore, the query log can be taken as a very valuable resource containing abundant relevance feedback data. Thus we can overcome the problem of lacking sufficient relevance judgment information in traditional relevance feedback technique. On the other side, our method has obvious advantage over pseudo relevance feedback - not only are the clicked documents the top-ranked documents, but also there is a further selection by the user. So document clicks are more reliable indications than those used in pseudo relevance feedback. Therefore, we can only expect better results.

The log based query expansion method has another three important properties. First, since the term correlations can be pre-computed offline, the initial retrieval phase is not needed anymore. Second, since the query logs contain the query sessions from different users, the term correlations can reflect the preference of most users. For example, if the majority of users use "windows" to search for information about Microsoft Windows product, the term "windows" will have much stronger correlations with the terms such as "Microsoft", "OS" and "software" rather than with the terms such as "decorate", "door" and "house". Thus the expanded query will lead to the documents about Microsoft Windows which are ranked higher than others. The similar idea has been used in several existing search engines, such as Direct Hit [5]. Our query expansion approach can naturally reach the same results. Third, the term correlations may evolve with the accumulation of user logs. The query expansion process can reflect the most users' query needs at a specific time.

2. PROBABILISTIC QUERY EXPANSION BASED ON QUERY LOGS

In this section, we introduce the details of the log based query expansion method. First, we will give a formal measurement of the similarity between query terms and document terms. Such kind of measurement is not feasible until a large amount of web logs are available nowadays. Our measurement results proved that there is a big gap between the query space and document space. Therefore some mechanisms are needed to bridge the gap, that is, to build up the relationships between query terms and document terms. Then we will discuss how to construct such a kind of term correlations by mining the query logs. Finally, we will show how to use these term correlations to select high-quality expansion terms.

2.1 Gap between the Query Space and the Document Space

Inconsistency between term usages in queries and documents is a well known problem in information retrieval. This is one of the very facts that motivate the use of query expansion. This problem was first observed by Furnas [6] in a more general context. It is even worse when the query is very short as is the case on the web.

But till now, no one precisely measures how different between the two word sets of documents and queries. It is difficult to determine the gap without user query logs. In order to acquire such a measurement, we collected two-month query logs (about 22 GB) from the Encarta search engine (http://encarta.msn.com), as well as the 41,942 documents in the Encarta website. From these logs we extracted 4,839,704 user query sessions. Below is an excerpt of the query sessions.

Queries	**IDs of clicked documents**
Trinidad and Tobago	761561556 761559363
Amish pacifism	761586809
Electric lights	761579230
Marion Jones	761562123
Ben Johnson	761562123
Spoils System	761551930
Indian removal act	761553925
Pecan tree pictures	761572753
New Mexico	761572098 761572098

Each document can be represented as a document vector $\{W_1^{(d)}, W_2^{(d)} .... W_n^{(d)}\}$ in the document space, where $W_i^{(d)}$ is the weight of the i^{th} term in a document and is defined by the traditional TF-IDF measure:

$$W_i^{(d)} = \frac{\ln(1 + tf_i^{(d)}) \times idf_i^{(d)}}{\sqrt{\sum \ln^2(1 + tf_i^{(d)}) \times \sum (idf_i^{(d)})^2}} \qquad (1)$$

$$idf_i^{(d)} = \ln \frac{N}{n_i} \qquad (2)$$

For each document, we can construct a corresponding virtual document in the query space by collecting all queries with clicks on the document. A virtual document is represented as a query vector $\{W_1^{(q)}, W_2^{(q)} ...... W_n^{(q)}\}$ where $W_i^{(q)}$ is the weight of the i^{th} term in the virtual document and also is defined by the TF-IDF measure.

To measure the gap between the query space and document space, we only need to measure the similarity between the document vector and its corresponding query vector. Specially, the similarity of each pair of vectors can be measured by using the following Cosine similarity:

$$Similarity = \frac{\sum_{i=1}^{n} W_i^{(q)} W_i^{(d)}}{\sqrt{\sum_{i=1}^{n} W_i^{2(q)}} \sqrt{\sum_{i=1}^{n} W_i^{2(d)}}} \qquad (3)$$

We noticed that many terms in the document space never or seldom be used in the users' queries. Thus many terms in the document vector do not appear or with very small weights in its corresponding query vector. Such a kind of mismatch could dramatically decrease the similarity between the two vectors. Therefore, we use the n most important common words in both vectors for the similarity calculation, where n is determined by the number of terms in the virtual document.

Figure 1 illustrates the final results of similarity values on the whole document collection. This figure shows that, in most cases, the similarity values of term usages between user queries and documents are between 0.1 and 0.4. Only very few documents have similarity values above 0.8. The average similarity value across the whole document collection is 0.28, which means the average internal angle between the query vector and the document vector is 73.68 degree. This is a rather big angle and indicates that there is really a broad gap between the query space and the document space.

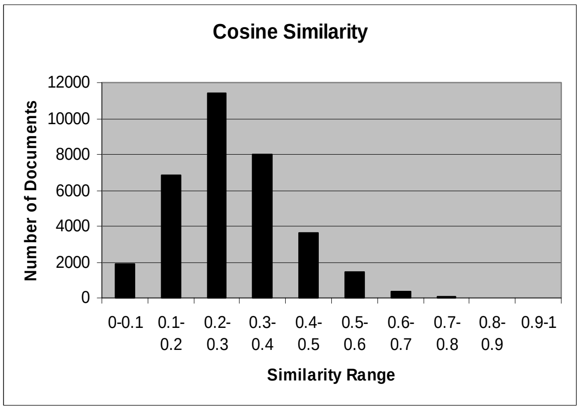

Figure 1 Similarity between the query terms and document terms

2.2 Correlations between Query Terms and Document Terms

Query sessions in the query logs provide an effective way to bridge the gap between the query space and the document space. Figure 2 shows that the correlations between the query terms and document terms can be established through the query sessions. In general, we assume that the terms in a query are correlated to the terms in the documents that the user clicked on. If there is at least one path between one query term and one document term, a probabilistic link is established between them. Thus we can obtain the probabilistic correlations between the terms in these two spaces (Figure 3).

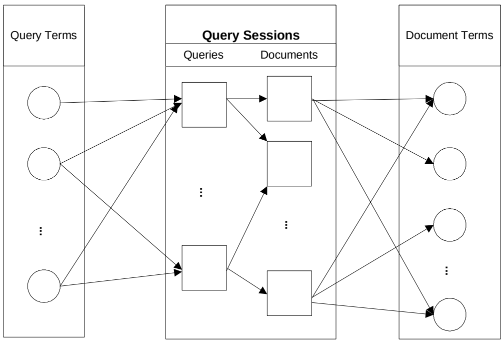

Figure 2 Query sessions, query terms and document terms

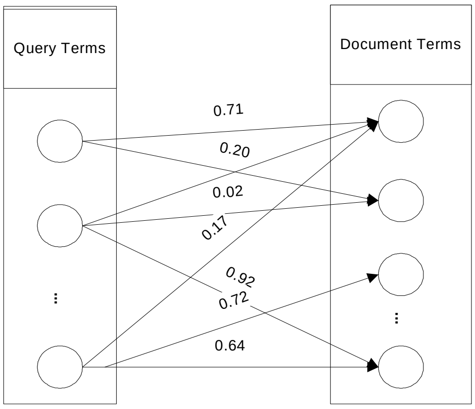

Figure 3 Probabilistic correlations between query terms and document terms

Now we discuss how to determine the degrees of correlations between terms. This is equal to calculating the conditional probabilities between them. In other words, the degree of the correlation between a query term and a document term is the conditional probability of a document term's appearance on condition that the query term is used. Let $w_j^{(d)}$ and $w_i^{(q)}$ be an arbitrary document term and a query term respectively. According to the Bayesian theorem, the conditional probability $P(w_j^{(d)} \mid w_i^{(q)})$ is defined as follows.

$$\begin{aligned}
P(w_j^{(d)} \mid w_i^{(q)}) &= \frac{P(w_j^{(d)}, w_i^{(q)})}{P(w_i^{(q)})} \\
&= \frac{\sum_{\forall D_k \in S} P(w_j^{(d)}, w_i^{(q)} \mid D_k) \times P(D_k)}{P(w_i^{(q)})} = \frac{\sum_{\forall D_k \in S} P(w_j^{(d)}, w_i^{(q)}, D_k)}{P(w_i^{(q)})} \\
&= \frac{\sum_{\forall D_k \in S} P(w_j^{(d)} \mid w_i^{(q)}, D_k) \times P(w_i^{(q)}, D_k)}{P(w_i^{(q)})} \\
&= \frac{\sum_{\forall D_k \in S} P(w_j^{(d)} \mid D_k) \times P(D_k \mid w_i^{(q)}) \times P(w_i^{(q)})}{P(w_i^{(q)})} = \sum_{\forall D_k \in S} P(w_j^{(d)} \mid D_k) \times P(D_k \mid w_i^{(q)})
\end{aligned} \tag{4}$$

S is a set of documents. A document is added into the set if and only if its document ID and the query term $w_i^{(q)}$ co-occur in at least one query session (that is, there is at least one user using the query term $w_i^{(q)}$ has clicked on the document). $P(D_k \mid w_i^{(q)})$ is the conditional probability of the document D_k being clicked in case that $w_i^{(q)}$ appears in the user query. $P(w_j^{(d)} \mid D_k)$ is the conditional probability of occurrence of $w_j^{(d)}$ if the document D_k is selected. It is noted that $P(w_j^{(d)} \mid w_i^{(q)}, D_k) = P(w_j^{(d)} \mid D_k)$ because the document D_k separates the query term $w_i^{(q)}$ from the document term $w_j^{(d)}$.

Obviously, $P(D_k \mid w_i^{(q)})$ can be statistically obtained from the query logs. $P(w_j^{(d)} \mid D_k)$ depends on the occurrences of $w_j^{(d)}$ in the document D_k , as well as the occurrences of the term $w_j^{(d)}$ in the whole document collection. So we use the following formulas to approximate $P(D_k \mid w_i^{(q)})$ and $P(w_j^{(d)} \mid D_k)$:

$$P(D_k \mid w_i^{(q)}) = \frac{f_{ik}^{(q)}(w_i^{(q)}, D_k)}{f^{(q)}(w_i^{(q)})} \tag{5}$$

$$P(w_j^{(d)} \mid D_k) = \frac{W_{jk}^{(d)}}{\max_{\forall t \in D_k} (W_{tk}^{(d)})} \tag{6}$$

Where

$f_{ik}^{(q)}(w_i^{(q)}, D_k)$ is the number of the query sessions in which the query word $w_i^{(q)}$ and the document D_k appear together.

$f^{(q)}(w_i^{(q)})$ is the number of the query sessions that contain the term $w_i^{(q)}$.

$W_{jk}^{(d)}$ is the normalized weight of the term $w_j^{(d)}$ in the document D_k , which is divided by the maximum value of term weights in the document D_k .

By combining the formulas (4), (5) and (6), we obtain the following formula to calculate $P(w_j^{(d)} \mid w_i^{(q)})$.

$$P(w_j^{(d)} \mid w_i^{(q)}) = \sum_{\forall D_k \in S} (W_{jk}^{(d)} \times \frac{f_{ik}^{(q)}(w_i^{(q)}, D_k)}{f^{(q)}(w_i^{(q)})}) \tag{7}$$

2.3 Query Expansion Based on Term Correlations

Our query expansion method is based on the probabilistic term correlations described above. When a new query comes, first, the terms in the query (with stop words being removed) are extracted. Then for every query term, all correlated document terms are selected based on the conditional probability obtained by the formula (7). By combining the probabilities of all query terms, we can get the joint probability for every document term by using the formula (8).

$$P(w_j^{(d)} \mid Q) = \ln(\prod_i (P(w_j^{(d)} \mid w_i^{(q)}) + 1)) \tag{8}$$

where Q stands for the new query.

Thus, for every query, we get a list of candidate expansion terms as well as the conditional probabilities between each term and the query. Then we can pick out the top-ranked terms as expansion terms and add them into the original query.

3. EXPERIMENTS AND RESULTS

In this section, we report the experimental results on the performance of the log based probabilistic query expansion method.

3.1 Data

We collected two-month query logs (about 22 GB) from the

Encarta website, from which 4,839,704 user query sessions are extracted. The documents collection is made up of 41,942 Encarta documents with various topics. In addition, the lengths of the documents vary greatly, from dozens of words to several thousand words.

Total 30 queries are used to conduct the experiments. Some queries are extracted randomly from the query logs. Some are selected from the TREC query set. Others are constructed manually. The queries in our experiments are very close to those employed by the real web users and the average length of all queries is 2.1 words. Figure 4 lists the 30 queries used in the experiments.

1 Java computer 2 nuclear submarine
3 Apple computer 4 Windows 5 fossil fuel
6 cellular phone 7 search engine
8 Six Day War 9 space shuttle
10 economic impact of recycling tires
11 China Mao Ze Dong 12 atomic bomb
13 Manhattan project 14 Sun Microsystems
15 Cuba missile crisis 16 motion pictures
17 Steve Jobs 18 pyramids 19 what is Daoism
20 Chinese music 21 genome project
22 Apollo program 23 desert storm
24 table of elements 25 Toronto film awards
26 Chevrolet truck 27 DNA testing
28 Michael Jordan 29 Ford 30 ISDN

Figure 4. The experimental query set

Relevant documents are judged according to the human assessors' manual selections. Thus, we can obtain the standard relevant document set for every query.

3.2 Word and Phrase Thesaurus

Encarta has well organized manual indexes in addition to automatically extracted index terms. In order to test our techniques in a general context, we did not use manual indexes or the existing Encarta searching engine for evaluation. Instead, we implement a vector space model as the baseline method in our experiments.

We build the word thesaurus out of all documents only excluding stop words. There are over 190,000 words in total in the word thesaurus.

We do not use traditional method to extract phrases from documents because we are more interested in the phrases in the query space. Therefore, we extract all N-grams from the query logs with occurrences bigger than 5 times and treat the N-grams as candidate phrases. Then we relocate these candidate phrases in the document corpus and filter out those not appearing in the documents. In the end we get a thesaurus containing over 13,000 phrases.

3.3 Quality of Expansion Terms

We examined the top 50 expansion terms for all the 30 queries to check the relevance of the expansion terms. As showed in the Table 1, compared to the local context analysis (LC Analysis), our log based query expansion method (Log Based) can achieve a 32.03% improvement of the quality of the expansion terms.

Table 1 Comparison on relevant percentage of expansion terms (Top 50 terms)

	LC Analysis (base)	Log Based	Improvement (%)
Relevant Terms (%)	23.27	30.73	+32.03

Figure 5 illustrates the expansion terms for the query "Steve Jobs" by our method. Some very good terms, such as "personal computer", "Apple Computer", "CEO" , "Macintosh", even "graphical user interface", "Microsoft" can be obtained by our techniques.

1.Apple 2. ***personal computer*** 3. ***Computers***

4***. personal computers*** 5. ***Apple Computer***

6. ***operating system*** 7. ***Newton***

8. ***graphical user interface*** 9. graphical user

10. ***Software*** 11. user interface

12. programming language 13. programming languages

14. ***computer*** 15. wozniak

16. CPU 17. operating systems

18. mainframe computer 19. personal

20. principia 21. jobs

22. ***CEO*** 23. company

24. computer systems 25. high-level

26. assembly language 27. machine language

28. computer system 29. ***Gates***

30. analog 31. circuit board

32. vice president 33. opticks

34. analytical engine 35. ***Microsoft***

36. jacquard 37. output devices

38. Halley 39. woolsthorpe

40. output device 41. Calculus

42. input devices 43. ***Lisa***

44. Pixar 45. first computer

46. Paul Allen 47. white light

48. ***Macintosh*** 49. slide rule 50. markkula

Figure 5 Expansion terms for "Steve Jobs"

3.4 Performance Comparison

Now we compare the retrieval performance of the log based query expansion method with the baseline (without query expansion) and the local context analysis method. Interpolated 11-point average precision is employed as the main metric of retrieval performance. Statistical paired t-test [7] is also used to determine significance of differences.

For the local context analysis, the default is to use 30 expansion terms, which include words and phrases, from 100 top-ranked documents for query expansion. The default δ is set to 0.1 here, as proposed in [18].

For the log based query expansion, we use 40 expansion terms. The expansion terms are extracted from top 100 relevant documents according to the query logs. Phrases appear in the query logs are assigned a parameter S , which is 10 in our experiments. The formula presented in section 4.1 is changed to:

$$P(w_j^{(d)} \mid w_i^{(q)}) = \sum_{\forall D_k \in S} (W_{jk}^{(d)} \times \frac{S \times f_{ik}^{(q)}(w_i^{(q)}, D_k)}{f^{(q)}(w_i^{(q)})}) \qquad (9)$$

Parameter S is a factor to promote the importance of phrases. Since queries are mostly far shorter than articles, occurrences of phrases should be stressed here. The alternative method is to employ idf since in general occurrences of phrases are less than that of words.

We now present the comparison results in Table 2 and plot the result on a chart as illustrated by Figure 6.

Table 2 A comparison of retrieval performance of Baseline, query expansion base on local context analysis (LC Exp), and log based query expansion (On Log Exp). All experiments are done with phrases included.

Recall	Baseline	LC Exp	On Log Exp
10	40.67	40.33(-0.82)	62.00(+52.46)
20	26.83	33.33(+24.22)	44.67(+66.46)
30	21.56	27.00(+25.26)	37.00(+71.65)
40	17.75	23.08(+30.05)	31.50(+77.46)
50	15.07	20.40(+35.40)	27.67(+83.63)
60	13.00	17.89(+37.61)	24.56(+88.89)
70	11.43	16.29(+42.50)	22.24(+94.58)
80	10.17	15.08(+48.36)	20.42(+100.82)
90	9.44	13.96(+47.84)	18.89(+100.00)
100	8.70	13.07(+50.19)	17.37(+99.62)
Average	17.46	22.04(+26.24)	30.63(+75.42)

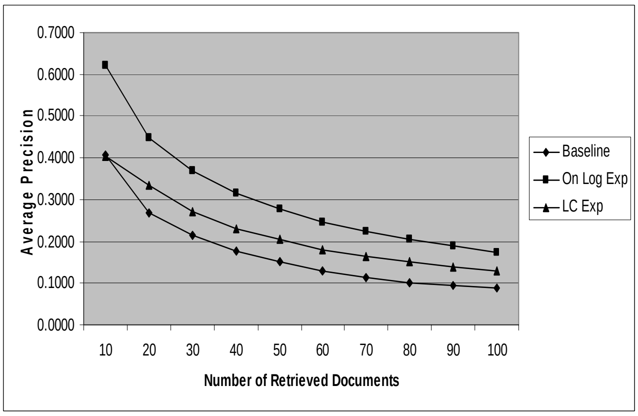

Figure 6 Average precision for Baseline, LC Exp and On Log Exp (with phrases)

Our log based query expansion performs well on the experiments. It has the maximum improvement of 100.82% (recall=80), and 75.42% improvement in average (p-value= 0.0000039585) over baseline, while the local context analysis achieve maximum improvement of 50.19% (recall=100) and 26.24% improvement in average (p-value= 0.018648254) over the baseline. The p-values of both tests show that both improvements are statistical significant. The p-values also indicate that our method can gain more significantly improvement over local context analysis. The average precision 17.46% of baseline is lower than that obtained by TREC experiments, which may be attributed to that queries in our experiments are much shorter than those in the TREC. This is closer to the real scenario on the Internet. In addition, local context analysis also offers large (26.24%) improvement compared to the baseline. That is a noticeable improvement, even slightly higher than the result reported in [18], which obtains 23.3% improvement on TREC3 and 23.5% improvement on TREC4. That indicates that query expansion is extremely important for short queries.

Log based query expansion also provides an average improvement of 38.95% compared to local context analysis, which is also statistically significant (p-value= 0.000493316). Generally, log based query expansion selects expansion terms in a relatively narrower but more concentrate style. In contrast, local context analysis searches expansion terms in the top-ranked retrieved documents and is more likely to add some irrelevant terms into the original query, which have side effects on retrieval performance.

3.5 Impact of Phrases

In [18], it is indicated that using noun phrases has little impact on retrieval performance. According to their experiments, the performance decreases by only 0.2% on TREC4 without using phrases. For the TREC queries, there is enough information to tradeoff the advantages of phrases since the queries are relatively long (7.5 words in average per query in TREC4). But for short queries, phrases are of crucial importance. It is reasonable because users often use phrases to represent their intentions. Without phrases, separate words in the query may lead to bad results. For example, “search engine” is a widely used phrase. But when we load word thesaurus only, it is parsed as “search” and “engine”. Then few of the retrieved documents are related to search engine, while most of them pertain to engines installed on motors, planes or vessels, etc. Our experiments show that the performance can be improved greatly when phrases are introduced into the query expansion and retrieval phases. On average, an 11.37% improvement can be obtained (see Table 3).

Table 3 Comparison of query expansion with phrases (Phrase) and without phrases (None Phrase)

Average Precision			
Recall	**None Phrase**	**Phrase**	**Improvements (%)**
10	53.00	62.00	16.98
20	40.67	44.67	9.84
30	32.56	37.00	13.65
40	28.67	31.50	9.88
50	25.47	27.67	8.64
60	22.78	24.56	7.81
70	20.43	22.24	8.86
80	18.63	20.42	9.62
90	17.04	18.89	10.87
100	15.80	17.37	9.92
Average	27.50	30.63	11.37

3.6 Impact of Number of Expansion Terms

In general, there is a peak point for a certain number of expansion terms to reach the best performance. Too many expansion terms not only consume more time in retrieval process, but also have side effects on retrieval performance.

We examine performance by using 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 expansion terms for retrieval. Table 4 and Figure 7 show the average precision obtained by different number of expansion terms. The best performances are obtained within the range of 40 and 60 terms. The performance drops when the number of expansion terms is larger than 70, which indicates that the terms beyond 70 are less relevant to the original query.

Table 4 Comparison of various number of expansion terms

Average 11-point Precision	
Exp Terms Count	**Precision**
10	0.271
20	0.294
30	0.303
40	0.306
50	0.306
60	0.308
70	0.304
80	0.304
90	0.302
100	0.296

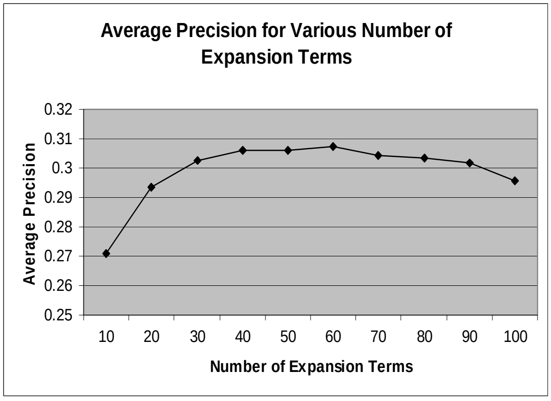

Figure 7 Impact of various number of expansion terms

4. RELATED WORK

One of the earliest global analysis techniques is term clustering [15], which groups document words into clusters based on their co-occurrences and then uses these groups to do query expansion. Other well-known global techniques include Latent Semantic Indexing [4], similarity thesauri [11], and PhraseFinder [8], etc.

The idea of local analysis can be traced back at least to a 1977 paper [1]. A more recent and well-known local analysis technique is relevance feedback [12, 14]. Local feedback mimics relevance feedback by assuming the top-ranked documents are relevant [3, 13]. 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 [10], clustering the top-ranked documents and removing the singleton clusters [9], clustering the retrieved documents but using the terms that best match the original query for expansion [2]. Local context analysis was proposed by Xu and Croft [18].

5. CONCLUSION

The proliferation of the World Wide Web prompts the wide application of search engines. However, short queries and inconsistency between users' query terms and document terms heavily affect the performance of existing search engines. Many automatic query expansion techniques have been proposed. These previous methods can solve the short query and term mismatch problem to some extent but they do not take into account exploiting the query logs, which widely reside in various websites, to involve users' accumulated interaction information in query expansion.

In this article, we present a novel method for automatic query expansion based on the query logs. The main thrust of this method is to establish probabilistic correlations between query terms and document terms through mining the query logs, which is an effective way to bridge the gap between the query space and the document space. Then for the newly coming queries, high-quality expansion terms can be selected from the document space on the basis of these probabilistic correlations. We tested this method on a data set which is similar to the real web environment. A series of experiments show that the log based method can achieve substantial improvement on performance, not only over the baseline method without expansion, but also the local context analysis which is one of the best query expansion methods.

Query expansion using query logs is only one application of web log mining. Other useful knowledge can be obtained through analyzing the users' behaviors recorded in the logs. We believe this is a very promising research direction.

6. ACKNOWLEDGEMENTS

We would like to thank Jian-Yun Nie for his kind review and Encarta group for the data set.

7. REFERENCES

[1] Attar, R. and Fraenkel, A.S. 1977. Local feedback in full-text retrieval systems. J. ACM 24, 3 (July), 397-417

[2] Buckley, C., Mitra, M., Walz, J. and Cardie, C. 1998. Using clustering and superconcepts within SMART. Proceedings of the 6th text retrieval conference (TREC-6), E. Voorhees, Ed. 107-124. NIST Special Publication 500-240

[3] Buckley, C., Salton, G., Allan, J. and Singhal, A. 1995. Automatic query expansion using SMART. TREC-3.

[4] Deerwester, S., Dumai, S.T., Furnas, G.W., Landauer, T.K. and Harshman, R. 1990. Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. 41,6, Pages 391-407

[5] Direct Hit website. http://www.directhit.com/

[6] Furnas, G.W., Landauer, T.K., Gomez, L.M. and Dumais, S.T. 1987. The vocabulary problem in human-system communication. Commun. ACM 30, 11 (Nov. 1987), Pages 964-971

[7] Hull, D. 1993. Using statistical testing in the evaluation of retrieval experiments. SIGIR'93, 1993

[8] Jing, Y. and Croft, W.B. 1994. An association thesaurus for information retrieval. RIAO'94, New York, NY

[9] Lu, A., Ayoub, M. and Dong, J. 1997. Ad hoc experiments using EUREKA. TREC-5, Pages 229-240.

[10] Mitra, M., Singhal, A. and Buckley, C. 1998. Improving automatic query expansion. SIGIR'98, 1998.

[11] Qiu, Y. and Frei, H. 1993. Concept based query expansion. SIGIR'93, Pittsburgh, PA.

[12] Rocchio. J. 1971. Relevance feedback in information retrieval. The Smart Retrieval system---Experiments in Automatic Document Processing. G. Salton. Ed. Prentice-Hall Englewood Cliffs. NJ. 313-323

[13] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 1999. Modern Information Retrieval. Pearson Education Limited, England, 1999.

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

[15] Sparck Jones, K. 1971. Automatic keyword classification for information retrieval. Butterworths, London, UK.

[16] Wen, J.R., Nie, J.Y. and Zhang, H.J. 2000. Clustering User Queries of a Search Engine. WWW10, May 1-5, 2001, Hong Kong.

[17] Xu, J. and Croft, W.B. 1996. Query expansion using local and global document analysis. 1996. SIGIR'96, 1996.

[18] Xu J. and Croft, W.B. 2000. Improving the effectiveness of information retrieval with local context analysis. ACM Transactions on Information Systems Vol.18, No.1, January 2000, Pages 79-112