# Query Expansion by Mining User Logs

Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma, *Member*, *IEEE*

**Abstract**—Queries to search engines on the Web are usually short. They do not provide sufficient information for an effective selection of relevant documents. Previous research has proposed the utilization of query expansion to deal with this problem. However, expansion terms are usually determined on term co-occurrences within documents. In this study, we propose a new method for query expansion based on user interactions recorded in user logs. The central idea is to extract correlations between query terms and document terms by analyzing user logs. These correlations are then used to select high-quality expansion terms for new queries. Compared to previous query expansion methods, ours takes advantage of the user judgments implied in user logs. The experimental results show that the log-based query expansion method can produce much better results than both the classical search method and the other query expansion methods.

**Index Terms**—Query expansion, user log, probabilistic model, information retrieval, search engine.

## 1 Introduction

In recent years, we have been witnessing the explosive growth of information on the World Wide Web. People are relying more and more on the Web for their diverse needs of information. However, the Web is an information hotpot where innumerous authors have created and are creating their Web sites independently. The vocabularies of the authors vary greatly. There is an acute requirement for search engine technology to help users 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 keyword-matching to determine the answers. Users usually describe their information needs by a few keywords in their queries, which are likely to be different from those index terms of the documents on the Web. This problem is general in Information Retrieval (IR) systems and has been documented before the popularization of the Web: *New or intermittent users often use the wrong words and fail to get the actions or information they want* [15]. As a consequence, in many cases, the documents returned by search engines are not relevant to the user information need. This raises a fundamental problem of term mismatch in information retrieval, which is also one of the key factors that affect the precision of the search engines.

Very short queries submitted to search engines on the Web amplify this problem: Many important words or terms may be missing from the queries. To solve this problem, researchers have investigated the utilization of query expansion techniques to help users formulate better queries. Query expansion involves supplementing the original query with additional words and phrases. There are two key aspects in any query expansion technique: the source from which expansion terms are selected and the method to weight and integrate expansion terms.

Manual query expansion has been studied by many researchers, for example, [1] and [17]. Manual query expansion demands user interventions. It is also required that the user is familiar with the online search system, the indexing mechanism, and the domain knowledge, which is generally not true for the users on the Web.

In this paper, we will focus on automatic query expansion. Current automatic query expansion techniques can be generally categorized into global analysis and local analysis.

A query expansion method based on global analysis usually builds a thesaurus to assist users reformulating their queries. A thesaurus can be automatically established by analyzing relationships among documents and statistics of term co-occurrences in the documents. From the thesaurus constructed in this way, one will be able to obtain synonyms or related terms given a user query. Thus, these related terms can be used for supplementing users' original queries.

Another group of techniques for query expansion is local analysis, which extracts expansion terms from a subset of the initial retrieval results. This subset may be determined directly by the user according to relevance judgments, or by the system (i.e., the top-ranked documents). Terms selected from them are added in a new query or their weights in the latter are increased [31]. Compared to the thesaurus-based expansion technique, local analysis is more query-oriented. Previous experiments have shown significant impact of local analysis on retrieval effectiveness. However, if the subset of documents is selected by the user, then we put a heavy burden on the user. If it is selected by the system, then it is questionable whether they are indeed relevant to the query; thus, the improvement on retrieval effectiveness is uncertain.

In this paper, we propose a new query expansion method based on user logs which record user interactions

- *H. Cui is with the Department of Computer Science, School of Computing, National University of Singapore, Singapore, 117543. E-mail: cuihang@comp.nus.edu.sg.*
- *J.-R. Wen and W.-Y. Ma are with Microsoft Research Asia, 3F Sigma Building, No. 49, Zhichun Rd. Haidian District, Beijing 100080, China. E-mail: {jrwen, wyma}@microsoft.com.*
- *J.-Y. Nie is with the Département d'informatique et Recherche Opérationnelle, Université de Montréal C.P. 6128, succursale Centre-ville Montreal, Quebec H3C 3J7 Canada. E-mail: nie@iro.umontreal.ca.*

*Manuscript received 15 July 2002; revised 15 Dec. 2002; accepted 6 Jan. 2003. For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number 118229.*

1041-4347/03/$17.00 © 2003 IEEE Published by the IEEE Computer Society

with the search systems. User logs are exploited so as to extract implicit relevance judgments they encode. In this approach, we assume that the documents that the user chose to read are "relevant documents." The log-based query expansion overcomes several difficulties of local analysis because we can extract a large number of user judgments from user logs, while eliminating the step of collecting feedbacks from users for ad hoc queries. Probabilistic correlations between terms in the user queries and the documents can then be established through user logs. With these term-term correlations, relevant expansion terms can be selected from the documents for a query. Our experiments show that mining user logs is extremely useful for improving retrieval effectiveness, especially for very short queries on the Web.

In this paper, we carry out a series of experiments to investigate the effects of our query expansion method on queries of different length. The experimental results on both long and short queries are presented in this article. As we will see, query expansion produces more significant improvements on short queries than on long queries.

The remainder of this paper is organized as follows: Section 2 describes the problem of inconsistency between query terms and document terms, which will motivate our approach. Our experimental result suggests a large difference between the terms used in queries and those in documents, therefore, the need in developing appropriate query expansion techniques for Web search. Section 3 reviews previous work on query expansion. Our log-based query expansion technique is described in detail in Section 4. Sections 5 and 6 describe the experiments comparing our method with local context analysis. Section 7 draws some conclusions.

## 2 Motivation

The problems of under-specification and inappropriate term usage in user queries are two motivations for studying query expansion. They are due to two facts: queries are often short, thus contain insufficient number of terms; and query terms are often inconsistent with (different from) those in the documents. In this section, we will examine these two facts with respect to a search engine on the Web.

It is generally observed that users on the Web typically submit very short queries to search engines and the average length of Web queries is less than two words [34]. A similar conclusion was drawn in [9]. We deduce that the very small overlap of the query terms and the document terms in the desired documents negatively affects the performance of Web searching.

In [15], it was observed that people use a surprisingly great variety of words when referring to the same thing and, thus, terms in user queries often fail to match the index terms contained in the relevant documents. It is even worse when the query is very short as on the Web. In this case, the chance of mismatching is much larger than for a long query.

In fact, we can view the term usages in the documents as forming a term space, that we call *document space*. The term usages in the queries form another term space—*query space*. The mismatching problem we just described comes from the inconsistency between the two spaces. This fact has often been hypothesized. However, no previous study has tried to measure the difference between the two spaces quantitatively. This measurement is difficult because the number of relevant judgments is always limited. With a large amount of user logs that we consider to encode relevance judgments, this becomes possible. In order to confirm the large difference between the two term spaces, we will measure the "similarity" between them. It is to be noted, however, that the resulting measure of similarity is an approximation. A true measure of similarity is only possible with real relevance judgments.

Our measurement is conducted with two-month user logs (about 22 GB) from the Encarta search engine (http://encarta.msn.com), as well as the 41,942 documents in the Encarta Web site. The user logs contains 4,839,704 user query sessions. Each query session consists of the query itself and its corresponding document clickthroughs (the documents on which the user clicked, see Section 4). Below is an excerpt of 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 |

We represent each document as a document vector $\left\{W_1^{(d)}, W_2^{(d)} \dots W_N^{(d)}\right\}$ in the document space, where $W_i^{(d)}$ is the weight of the $i$th term in a document and it is defined by the traditional TF*IDF measure:

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

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

where $tf_i^{(d)}$ is the frequency of the $i$th term in the document $D$, $N$ the total number of documents in the collection, and $n_i$ the number of documents containing the $i$th term. For each document, we can construct a corresponding virtual document in the query space by collecting and counting all the terms, excluding stopwords, in the queries for which the document has been selected and clicked on by the user. A virtual document is represented as a query vector $\left\{W_1^{(q)}, W_2^{(q)} \dots W_N^{(q)}\right\}$, where $W_i^{(q)}$ is the weight of the $i$th term in the virtual document and it is also defined by the TF*IDF measure.

The similarity between the two vectors is calculated and it is assumed to reflect the similarity between the query space and document space we measure. Specially, the similarity of each pair of vectors is calculated using the following Cosine formula:

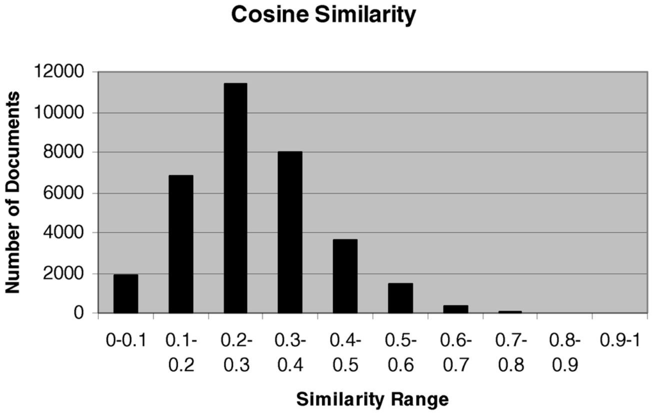

Fig. 1. Similarity between the query terms and the document terms.

$$Similarity = \frac{\sum_{i=1}^{N} W_i^{(q)} W_i^{(d)}}{\sqrt{\sum_{i=1}^{N} (W_i^{(q)})^2} \sqrt{\sum_{i=1}^{N} (W_i^{(d)})^2}}. \tag{3}$$

We notice that many terms in the document space never or seldom appear in the users' queries. Thus, the query vector created is much shorter (with less nonzero terms) than a document vector. This artifact will dramatically decrease the similarity between the two vectors if all the terms are used in the measurement. To obtain a fairer measure, we only use the $n$ highest ranking words in each document vector for the similarity calculation, where $n$ is the number of terms in the corresponding query vector.

Fig. 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 result suggests that there is indeed a large gap between the query space and the document space. It is thus very difficult to retrieve the desired documents with a direct keyword matching approach. It is important to find ways to narrow the gap or to bridge the two spaces in order to improve retrieval effectiveness.

## 3 Review of Previous Work on Automatic Query Expansion

In this section, let us review some previous approaches to query expansion. The existing state-of-the-art query expansion approaches can be classified mainly into two categories—techniques based on global analysis, which obtains expansion terms on the statistics of terms in the whole corpus, and local analysis, which extracts expansion terms from a subset of the search results.

### 3.1 Global Analysis

In this section, we only review the approaches that exploit term co-occurrences in documents. We do not analyze the approaches that use a manual thesaurus (e.g., WordNet [22]). One can refer to [33] for some examples of utilization of such a resource for query expansion.

Global analysis is one of the first techniques to produce consistent and effective improvements through query expansion. The basic idea of global analysis is to use the context of a term to determine its similarity with other terms. Global analysis selects expansion terms on the basis of the information on the whole document set. It builds a set of statistical term relationships which are then used to expand queries.

One of the earliest global analysis techniques is term clustering [20], [32]. Queries are simply expanded by adding similar terms that are grouped into the same cluster according to term co-occurrences in documents.

Qiu and Frei [24] presented a query expansion model using a global similarity thesaurus. Another work based on a global statistical thesaurus is [10], which first clusters documents and then selects low-frequency terms to represent each cluster. PhraseFinder [19] is a component of the INQUERY system that creates an association thesaurus. The phrases selected by PhraseFinder are used in query expansion.

Latent Semantic Indexing [12] can also be viewed as a kind of query expansion. In its reduced dimensional space, implicit correlations among terms can be discovered and employed in expanding original queries.

Generally, global analysis requires corpus-wide statistics, such as statistics of co-occurrences of pairs of terms, resulting in a matrix of similarities between terms or a global association thesaurus. Although the global analysis techniques are relatively robust, the corpus-wide statistical analysis consumes a considerable amount of computing resources. Moreover, since it focuses only on the document side and does not take into account the query side, global analysis only provides a partial solution to the term mismatching problem.

### 3.2 Local Analysis

Different from global analysis, local analysis uses only a subset of documents that is returned with the given query. The result is thus more focused on the given query than global analysis. Local analysis techniques are grouped into two categories: approaches based on user feedback information and approaches based on information derived from a subset of the returned documents.

#### *3.2.1 Relevance Feedback*

Relevance feedback is a straightforward strategy for reformulating queries. In a relevance feedback cycle, the user is presented with a list of initial results. After examining them, the user marks those documents he or she considers relevant. The original query is expanded according to these relevant documents. The expected result is that the next round of retrieval will move toward the relevant documents and away from nonrelevant documents.

Early experiments with the Smart system [30] and later experimental results using a probabilistic model [25] indicate improvements in effectiveness with relevance feedback for small collections.

Rocchio performed query reformulation using vector space model and obtained significantly positive results [27]. Salton and Buckley [31] did experiments on six test collections to compare various relevance feedback methods.

Their work mainly consisted of term reweighting and query expansion.

Typically, expansion terms are extracted from the relevant documents judged by the user. Relevance feedback can achieve very good performance if the user provides sufficient and correct relevance judgments. Unfortunately, in a real search context, users usually are reluctant to provide such relevance feedback.

#### 3.2.2 Local Feedback

To overcome the difficulty due to the lack of sufficient relevance judgments, local feedback, also known as blind feedback or pseudofeedback, is commonly used in IR. Local feedback mimics relevance feedback by assuming that the top-ranked documents are relevant [4]. Expansion terms are extracted from the top-ranked documents to formulate a new query for a second-cycle retrieval.

Local feedback has been proven effective in previous TREC experiments. In some cases, it outperforms global analysis [6], [13], [14], [26]. Nevertheless, this method can hardly overcome its inherent drawback: If a large fraction of the top-ranked documents are actually irrelevant, then the words added into the query (drawn from these documents) are likely to be unrelated to the topic and as a result, the quality of the retrieval using the expanded query is degraded. Therefore, the effect of pseudofeedback strongly depends on the quality of the initial retrieval.

In recent years, many improvements for local feedback have been proposed. Mitra et al. [23] suggested improving query expansion by refining the set of documents used in feedback with Boolean filters and proximity constraints. Clustering the top-ranked documents and removing the singleton clusters are techniques used in [21] in order to concentrate on large groups of relevant documents for query expansion. Buckley et al. [5] employed clustering to identify concepts. More recently, Carpineto et al. [7] presented a method of weighting and selecting expansion terms using Information Theory. To enhance the reliability of pseudorelevance feedback (PRF), *Flexible PRF* was proposed in [29], which varies the number of expansion terms according to the number of documents retrieved.

Recently, Xu and Croft [37], [38] proposed a local context analysis method, which applies the measure of global analysis to the selection of query terms in local feedback. From the top-ranked documents, noun groups are selected according to their co-occurrences with the query terms. In this way, the local context analysis method can solve the problem of insufficient statistical data of local analysis to some extent. However, local context analysis is based on the hypothesis that a frequent 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 hypothesis, but not always true, as shown by our examination on the gap between the document and query spaces. This is precisely the problem we will address by exploiting user logs for query expansion.

## 4 Log-Based Query Expansion

To deal with the mismatching problem at its source, i.e., the inconsistency problem between the terms used in the documents and those used in the queries, a possible way is to create relationships between the two sets of terms. User logs provide a resource exploitable for this end.

### 4.1 Principle of Using User logs

We observe that many search engines have accumulated a large amount of user logs from which we can know what the query is and what the documents users have selected to read. These user logs provide valuable indications to understand the kinds of documents the users intend to retrieve by formulating a query with a set of particular terms. There has been some work on mining user logs to enhance Web searching. Beeferman and Berger [2] exploited "clickthrough data" in clustering URLs and queries using graph-based iterative clustering technique. Wen et al. [34] used a similar method to cluster queries according to user logs in order to find Frequently Asked Questions (FAQs). These FAQs are then used to improve the effectiveness of question answering.

In this study, we further extend the previous utilizations of user logs by trying to extract relationships between query terms and document terms. These relationships are then used for query expansion. Thus, our work may be viewed as a trial to construct a live thesaurus that bridges the document and the query spaces. The general principle is: If queries containing one term often lead to the selection of documents containing another term, then we consider that there is a strong relationship between the two terms. This principle is an extension to that exploiting term co-occurrences. In previous approaches, term co-occurrences are observed within documents. The term relationships extracted from them are those between the terms used by the same authors. Therefore, we can see them as relationships within the document space. As we explained earlier, an important factor of the mismatching problem is the lack of relationships between the document space and the query space. There is an acute need to create a bridge between them. The idea of exploiting user logs precisely aims to create such a bridge between the two spaces.

To exploit this principle, our first task is to extract query sessions from a large set of noisy log data. The query sessions we extract are defined as follows:

$$\text{session} :=< \text{query text} > [\text{clicked document}]*$$

Each session contains one query and a set of documents which the user clicked on (which we will call *clicked documents*). 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 strongly related to the terms of the queries. Thus, some probabilistic correlations between query terms and document terms can be established based on the user logs.

One important assumption behind this method is that the clicked documents are "relevant" to the query. At the first glance, this assumption may appear too strong. However, although the clicking information is not as accurate as explicit relevance judgment in traditional relevance feedback, the user's choice does suggest a certain degree of relevance. Typically, upon getting a list of documents, many users do not select resulting documents

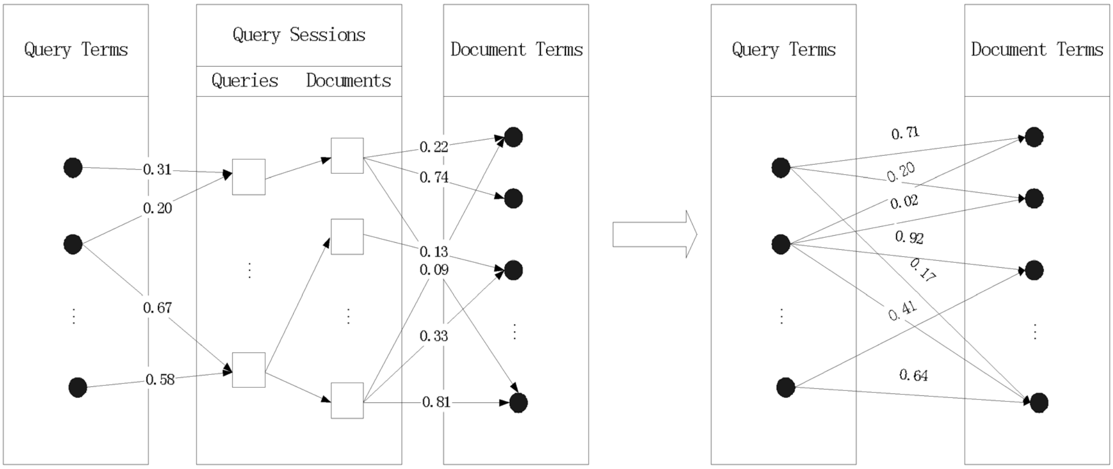

Fig. 2. Establishing correlations between query terms and document terms via query sessions.

randomly. They have a rough idea of what the documents are about from their titles and snippets. In most of the cases, they click and read those documents which are the most similar to what they have in mind. Therefore, these clicked documents do have some relationship with the queries they submit. Of course there are exceptions, such as an error click or a sudden shift in the user's intention. But, in the long run with a large amount of log data, the click-through records allow us to find strong correlations among terms from a statistical point of view. Similar observation has been made in [34]. On the whole, user logs can be viewed as a very reliable resource containing abundant implicit relevance judgments.

### 4.2 Characteristics of Log-Based Query Expansion

In a more general sense, the log-based query expansion method may be viewed as a special case of local analysis because its expansion terms are derived from a subset of the documents. However, it is enhanced by human judgments: Not only the clicked documents are usually top-ranked documents, but also they have been selected by the users. This method thus has several advantages over relevance feedback and pseudorelevance feedback.

Recall that the factor which limits the application of relevance feedback is the unavailability of user relevance judgments in a multiple-query process. Users tend to mark only a few, if any, documents when presented with a list of resulting documents. In addition, this feedback information can be exploited only once. Once the query is changed, the same feedback process is to be started again. Log-based query expansion collects and analyzes all users' historical relevance judgments as a whole without intervention of users. We benefit from abundant records of "voted" documents, while the biases or errors in a single round of feedback can be minimized. Thus, we can overcome the problem of lacking sufficient relevance judgments in previous local feedback techniques.

On the other hand, compared to the pseudorelevance feedback, our method has an obvious advantage: Not only are the clicked documents part of the top-ranked documents, but also there is a further selection by the user. Because document clicks are more reliable indications than top-ranked documents used in pseudorelevance feedback, log-based query expansion is expected to be more robust and accurate than the former.

The log-based query expansion method has three other important properties. First, since the term correlations can be precomputed offline, the initial retrieval phase of pseudorelevance feedback is not needed anymore. Second, since user logs contain query sessions from different users, the term correlations can reflect the preference of the majority of the users. For example, if the majority of the users use "windows" to search for information about Microsoft Windows product, the term "windows" will have stronger correlations with the terms such as "Microsoft," "OS," and "software," than with the terms such as "decorate," "door," and "house." Thus, the expanded query will result in a higher ranking for the documents about Microsoft Windows, which corresponds to the intentions of most users. The similar idea has been used in several existing search engines, such as Direct Hit (http://www.directhit.com). Our query expansion approach can produce the same results. Third, the term correlations may evolve along with the accumulation of user logs. Hence, the query expansion process can reflect updated user interests at a specific time.

### 4.3 Correlations between Query Terms and Document Terms

Query sessions in the user logs provide a possible way to bridge the gap between the query space and the document space. As illustrated in Fig. 2, weighted links can be created between the query space (all the query terms) and the query sessions, as well as between the document space (all the document terms) and the 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 link is created between them. Thus, the correlations between the query terms and document terms can be measured by investigating the weights of the links constituting the path between them. By analyzing a large number of such links, we can obtain a new matrix storing probabilistic correlations between the terms in these two spaces (the right part of Fig. 2). This is similar, in principle, to building a term-term similarity thesaurus in global analysis as in [36]. However, it benefits from the additional user judgments.

Let us now discuss how to determine the degrees of correlation between terms. We define these degrees as the conditional probabilities between terms, i.e., $P(w_j^{(d)}|w_i^{(q)})$ for any document term $w_j^{(d)}$ and any query term $w_i^{(q)}$. The probability $P(w_j^{(d)}|w_i^{(q)})$ can be determined as follows (where $S$ is a set of clicked documents for queries containing the query term $w_i^{(q)}$):

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

We assume that $P(w_j^{(d)}|w_i^{(q)}, D_k) = P(w_j^{(d)}|D_k)$. This means that the document $D_k$ separates the query term from the document term $w_j^{(d)}$. Therefore,

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

$P(D_k|w_i^{(q)})$ is the conditional probability of the document $D_k$ being clicked when $w_i^{(q)}$ appears in the user query. $P(w_j^{(d)}|D_k)$ is the conditional probability of occurrence of $w_j^{(d)}$ if the document is selected. $P(D_k|w_i^{(q)})$ and $P(w_j^{(d)}|D_k)$ can be estimated, respectively, from the user logs and from the frequency of occurrences of terms in documents as follows:

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

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

where

- $f_{ik}^{(q)}(w_i^{(q)}, D_k)$ is the number of the query sessions in which the query term $w_i^{(q)}$ and the document $D_k$ appear together.
- $f^{(q)}(w_i^{(q)})$ is the number of the query sessions that contain the term $w_i^{(q)}$.
- $P(w_j^{(d)}|D_k)$ is the normalized weight of the term $w_j^{(d)}$ in the document $D_k$, which is divided by the sum of all term weights in the document $D_k$.

By combining (4), (5), and (6), we obtain the following formula for $P(w_j^{(d)}|w_i^{(q)})$:

$$P(w_j^{(d)}|w_i^{(q)}) = \sum_{\forall D_k \in S} \left( P(w_j^{(d)}|D_k) \times \frac{f_{ik}^{(q)}(w_i^{(q)}, D_k)}{f^{(q)}(w_i^{(q)})} \right). \tag{7}$$

### 4.4 Query Expansion Based on Term Correlations

Equation 7 describes how to calculate the chance of a document term being selected as an expansion term given a query term. We also need to determine the relationship of a document term to the whole query in order to rank it.

For this, we use an idea similar to that of [24], i.e., we select expansion terms according to their relationship to the whole query. The relationship of a term to the whole query is measured by the following cohesion calculation:

$$CoWeight_Q(w_j^{(d)}) = \ln\left(\Pi_{w_t^{(q)} \in Q}\left(P\left(w_j^{(d)}|w_t^{(q)}\right) + 1\right)\right) \tag{8}$$

which combines the relationships of the term to all the query terms.

On the whole, log-based query expansion takes the following steps to expand a new query $Q$:

1. Extract all query terms (eliminating stopwords) from $Q$.
2. Find all documents related to any query term in query sessions.
3. To each document term in these documents, use (8) to calculate its evidence of being selected as an expansion term according to the whole query.
4. Select $n$ document terms with the highest cohesion weight and formulate the new query $Q'$ by adding these terms into $Q$.
5. Use $Q'$ to retrieve documents in a searching system.

## 5 Experimental Data and Methodology

Before illustrating the experimental results, let us first describe the test data used.

### 5.1 Data

Due to the characteristics of our query expansion method, we cannot conduct experiments on standard test collections such as the TREC[1] data since they do not contain user logs that we need. To deduct term-term correlations, we use the same two-month user logs from the Encarta Web site as described in Section 2, which contains 4,839,704 user query sessions. With respect to documents set, we collected 41,942 documents from the Encarta Web site to form the test corpus. Diverse topics are covered by these articles with greatly varying lengths, from dozens of words to several thousand words. In user logs, each document bears a large number of queries with which users have clicked on that document. This ensures that we have sufficient click-through information to establish meaningful probabilistic correlations among terms in the two spaces. In addition, this data set can reflect the impact of our query expansion technique for searches on the Web since it is obtained from a real search engine.

We focus on using query expansion to counter the effect of short queries on the Web. Xu and Croft [38] conducted experiments on very short queries, in which the results showed that query expansion can produce even larger

[1] http://trec.nist.gov/.

TABLE 1
List of Queries in Both the Long Query Set and the Short Query Set

| ID | Short Queries | Long Queries |
|---|---|---|
| 1 | Java | Tell me something about the computer language-Java and what its program is like. |
| 2 | nuclear submarine | nuclear submarine and ballistic missiles in super powers |
| 3 | Apple computer | Macintosh-one of the computers produced by Apple |
| 4 | Windows | What are the features of Windows that Microsoft bring us? |
| 5 | fossil fuels | Find information about fossil fuels, such as oil, and coal. |
| 6 | cellular phone | Which companies are producing cellular phones or mobile phones? |
| 7 | search engine | What is the role of search engine for the Web? |
| 8 | Six Day War | The Six Day War between Israel and Arab. |
| 9 | space shuttle | The difference between a space shuttle and a space station. |
| 10 | recycling tires | what is the economic impact of recycling tires |
| 11 | China Mao Ze Dong | The communist leader of China-Mao Ze Dong. |
| 12 | atomic bomb | Give me something on atomic bomb and other nuclear weapons. |
| 13 | Manhattan Project | What is the result of Manhattan Project during World War II? |
| 14 | Sun Microsystems | Find out who founded Sun Microsystems and what its main operating system is. |
| 15 | Cuba missile crisis | The Cuba Missile Crisis between USSR and United States. |
| 16 | motion pictures | The history of the motion pictures, or the film. |
| 17 | Steve jobs | Is Steve Jobs the CEO of Apple Computer? |
| 18 | pyramids | I want to know something about the pyramids in Egypt and other countries. |
| 19 | Bill Gates | Is Allan Paul a friend of Bill Gates? |
| 20 | Chinese music | The difference between Chinese music and classic western music. |
| 21 | genome project | Why is the Genome Project so crucial for humans? |
| 22 | Apollo program | Who landed on the moon in the Apollo lunar program? |
| 23 | Desert Storm | Who is the commander of the military operation-Desert Storm? |
| 24 | table of elements | The Table of Elements in the history of chemistry. |
| 25 | Toronto Film Awards | Find documents that discuss the Toronto Film Festival awards. |
| 26 | Chevrolet truck | Find documents that address the types of Chevrolet trucks available. |
| 27 | DNA testing | DNA testing for crimes and cancer. |
| 28 | Michael Jordan | Michael Jordan in NBA matches. |
| 29 | Ford | The history and present of Ford Motors. |
| 30 | ISDN | How to use ISDN to access the Internet? |

improvements on short queries than on long queries. We compiled two sets of queries in order to see how query expansion affects retrieval results on short queries and long queries. In order to test our method on a more general basis, some queries are extracted randomly from the user logs. Some others come from the TREC query set. Yet, another subset of queries is added manually by us. Table 1 shows all the 30 queries in both short and log versions used in our experiments.

The short queries in our experiments are very close to those employed by the real Web users and the average length of these queries is 2.0 words. The average length of the long queries is 4.8 keywords (excluding the stopwords). Though it is still shorter than the average length of most TREC queries, we consider that it reflects the real situation on the Web since few users use over five keywords to express their information needs.

We used three human assessors to build the relevance judgments. Relevant documents for each query were judged according to the human assessors' manual selections, and standard relevant document sets were prepared for all of the 30 queries. Assessors had no knowledge of the testing methods, but made decisions with the assistance of a basic searching system. To solve their disagreements when they occurred, the assessors discussed them together. All judgments from the assessors constituted a reference set. We run all experiments in a batch mode according to the relevance judgment set.

### 5.2 Word and Phrase Thesaurus

Encarta has well-organized manual indexes in addition to automatically extracted index terms. In order to test our technique in a general context, we do not use the manual indexes and the existing Encarta search engine which exploits it for our evaluation. Instead, we implement a vector space model as the baseline method in our experiments.

We do not use traditional methods to extract phrases from documents because we are more interested in the phrases in the query space. Therefore, we extract all sequences of N-grams, where N is the number of nontrivial terms in a query, from the user logs with occurrences higher than 5. These N-grams are treated as candidate phrases. Then, we locate the 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, which are used as additional indexes. When using phrases and single words together, our system always gives priority to phrases.

### 5.3 Evaluation Methodology

In order to evaluate our log-based query expansion method, we will compare its performance not only with that of the original queries, but also with that of local context analysis. We employ interpolated 11-point average precision as the main metric of performance. Statistical t-test [18] is used to indicate whether an improvement is statistically significant. A p-value less than 0.05 is deemed significant.

TABLE 2
A Comparison of Retrieval Performance in Average Precision (%) for Long Queries between Baseline, Local Context Analysis (LC Exp), and Log-Based Query Expansion (On Log Exp)

| Recall | Baseline | LC Exp | On Log Exp |
|---|---|---|---|
| 10 | 46.67 | 41.67(-10.71%) | 57.67(+23.57%) |
| 20 | 31.17 | 34.00(+9.09%) | 42.17(+35.29%) |
| 30 | 25.67 | 27.11(+5.63%) | 34.89(+35.93%) |
| 40 | 21.67 | 23.50(+8.46%) | 30.50(+40.77%) |
| 50 | 18.40 | 20.60(+11.96%) | 26.93(+46.38%) |
| 60 | 16.33 | 18.33(+12.24%) | 24.17(+47.96%) |
| 70 | 14.52 | 16.67(+14.75%) | 21.76(+49.84%) |
| 80 | 13.33 | 15.54(+16.56%) | 19.79(+48.44%) |
| 90 | 12.33 | 14.37(+16.52%) | 18.22(+47.75%) |
| 100 | 11.37 | 13.53(+19.06%) | 16.83(+48.09%) |
| Average | 21.15 | 22.53(+6.56%) | 29.29(+38.53%) |

TABLE 3
A Comparison of Retrieval Performance in Average Precision (%) for Short Queries between Baseline, Local Context Analysis (LC Exp), and Log-Based Query Expansion (On Log Exp)

| Recall | Baseline | LC Exp | On Log Exp |
|---|---|---|---|
| 10 | 40.67 | 45.00(+10.66%) | 62.33(+53.28%) |
| 20 | 27.00 | 32.67(+20.99%) | 44.33(+64.20%) |
| 30 | 20.89 | 26.44(+26.60%) | 36.78(+76.06%) |
| 40 | 17.25 | 22.33(+29.47%) | 31.33(+81.64%) |
| 50 | 14.53 | 19.67(+35.32%) | 27.27(+87.61%) |
| 60 | 12.39 | 17.78(+43.50%) | 24.33(+96.41%) |
| 70 | 10.90 | 16.33(+49.78%) | 21.95(+101.31%) |
| 80 | 9.63 | 15.08(+56.71%) | 20.04(+108.23%) |
| 90 | 8.81 | 13.93(+57.98%) | 18.56(+110.50%) |
| 100 | 8.03 | 13.13(+63.49%) | 17.07(+112.45%) |
| Average | 17.01 | 22.24(+30.72%) | 30.40(+78.71%) |

Terms are weighted using TF*IDF measure in our retrieval system. Both the original and the expanded queries are evaluated by the same retrieval system, making it possible to compare the effects of query expansion.

For local context analysis, we use 30 expansion terms (including words and phrases) from 100 top-ranked documents for query expansion. The smoothing factor $\delta$ in local context analysis is set to 0.1, as suggested by [38]. For the log-based query expansion, we use 40 expansion terms.

We notice that the occurrences of phrases are far less than those of words. This creates an unbalance between the weights we assigned to word correlations and to phrase correlations. In order to create a better balance, the probability associated with a phrase correlation is multiplied by a factor $S$ because phrases are less ambiguous than words ($S$ is set to 10 in our experiments). The formula used to measure phrase correlations is modified from (7) to the following one:

$$P(T_j^{(d)}|T_i^{(q)}) = \sum_{\forall D_k \in S} \left( P(T_j^{(d)}|D_k) \times \frac{S \times f_{ik}^{(q)}(T_i^{(q)}, D_k)}{f^{(q)}(T_i^{(q)})} \right), \quad (9)$$

where $T_j^{(d)}$ and $T_i^{(q)}$ are, respectively, a document phrase and a query phrase. In addition, the above results of (7) and (9) should be divided by the sum of all $P(w_j^{(d)}|w_i^{(q)})$ and $P(T_j^{(d)}|T_i^{(q)})$ in order to satisfy the requirement of the probabilistic framework.

# 6 Experimental Results

## 6.1 Performance Comparison

We now present the experimental results of the local context analysis and the log-based query expansion method. Results with the original queries without expansion are used as the baseline. All the experiments are carried out with both words and phrases. The results with the long queries and the short queries are presented, respectively, in Table 2 and Table 3.

We see that our log-based query expansion performs very well on both query sets. On the long query set, the log-based query expansion method brings an average improvement of 38.53 percent in precision (p-value = 0.000077) over the baseline, while the local context analysis achieves an average improvement of 6.56 percent in precision (p-value = 0.33) over the baseline. The p-value suggests that the log-based query expansion gains a statistically significant improvement over the original queries. It is to be noted that the log-based query expansion also provides an average improvement of 30.00 percent compared to local context analysis, which is also statistically significant (p-value = 0.0017). In general, we observe that log-based query expansion selects more accurate expansion terms than local context analysis due to the exploitation of user judgments. 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, thus introducing some undesirable side-effects on retrieval performance.

The results shown in Table 3 advocate our conjecture that our query expansion approach is even more useful for short queries than for long queries. There is a dramatic change in the performances of both local context analysis and the log-based query expansion method when short queries are expanded. The log-based query expansion offers an average improvement of 78.71 percent (and maximum improvement of 112.45 percent) in comparison with the original queries. The p-value for this augment is 0.0000056 which indicates its statistical significance. Local context analysis boosts the average precision to 22.24 percent, which is 30.72 percent better than the baseline (p-value = 0.018) (compared to only 6.56 percent improvement gained on the long query set). All these results suggest that query expansion is of extreme importance for short queries. According to our observation that less than two words are used in most user queries in the Encarta logs, we come to the conclusion that query expansion may improve the effectiveness of search engines which deals with short queries.

It is interesting to compare the results of the query expansion techniques on both query sets. With the local context analysis, the results with the long queries are slightly better than those with short queries, with an improvement of 1.33 percent. However, the results obtained by the log-based query expansion on the long queries are 3.64 percent worse than their counterparts for short queries.

TABLE 4
Comparison of Average Precision (%) obtained by Log-Based Query Expansion with Phrases (Phrase) and without Phrases (No Phrase) on the Long Query Set

| Recall | No Phrase | Phrase |
|---|---|---|
| 10 | 53.00 | 57.67 (+8.81) |
| 20 | 39.50 | 42.17 (+6.75) |
| 30 | 32.78 | 34.89 (+6.44) |
| 40 | 28.50 | 30.50 (+7.02) |
| 50 | 25.13 | 26.93 (+7.16) |
| 60 | 22.44 | 24.17 (+7.67) |
| 70 | 20.33 | 21.76 (+7.03) |
| 80 | 18.67 | 19.79 (+6.03) |
| 90 | 17.19 | 18.22 (+6.03) |
| 100 | 16.03 | 16.83 (+4.99) |
| Average | 27.36 | 29.29 (+7.08) |

TABLE 5
Comparison of Average Precision (%) obtained by Log-Based Query Expansion with Phrases (Phrase) and without Phrases (No Phrase) on the Short Query Set

| Recall | No Phrase | Phrase |
|---|---|---|
| 10 | 53.00 | 62.33 (+17.61) |
| 20 | 39.17 | 44.33 (+13.19) |
| 30 | 31.67 | 36.78 (+16.14) |
| 40 | 27.67 | 31.33 (+13.25) |
| 50 | 24.67 | 27.27 (+10.54) |
| 60 | 22.06 | 24.33 (+10.33) |
| 70 | 19.76 | 21.95 (+11.08) |
| 80 | 18.00 | 20.04 (+11.34) |
| 90 | 16.44 | 18.56 (+12.84) |
| 100 | 15.37 | 17.07 (+11.06) |
| Average | 26.78 | 30.40 (+13.52) |

This may suggest that our method can select expansion terms for short queries that are even better than those used in the long queries to describe the information needs. Globally, with query expansion, the performances for short and long queries are similar. This confirms that query expansion is an effective way to reduce the difference between short and long queries.

### 6.2 Impact of Phrases

Experiments on noun phrases in [38] showed that the local context analysis can achieve a small improvement with phrases. However, they only tested it with long queries. We believe that this impact can be even larger for short queries. In fact, even if a word-based representation is not precise, in a long query, this imprecision is compensated by the large number of words in the query. The whole set of query words together may give a quite precise description of the information need. However, this is not the case for short queries. For short queries, the user's intention can be expressed more accurately with phrases because phrases are inherently less ambiguous than single words. We conduct experiments of the log-based query expansion with and without phrases. The results are shown in Table 4 and Table 5. The results confirm our expectation just described. The improvement gained with phrases on the short queries is almost twice of that obtained with phrases on the long queries.

Similar to the retrieval process, query expansion is also affected by the ambiguity of the terms in original queries. The use of phrases can help reduce the ambiguity of query terms, thus allow query expansion to extract more relevant expansion terms. For example, for the short version of the query #8 "Six Day War" (see Table 1), each word is common and appears in many documents irrelevant to this query. If it is parsed as three single words, many irrelevant documents will be found. However, when it is presented as a phrase, the concept represented by it becomes unambiguous and it can match less irrelevant documents; so, the retrieval effectiveness can be improved. In comparison, given the long version of this query, if the three words are supplemented by the words "Israel" and "Arab," then they describe together a more precise meaning, leading to more relevant documents. So, even though phases are not recognized in a long query, the impact is less dramatic than for a short query.

Our other results (that are not listed here) show that, if we use phrases in the baseline method, the performance of this latter can also be improved by 2.35 percent and 8.95 percent, respectively, on the long and the short queries. Integrating phrases into the local context analysis can achieve improvements of 8.21 percent and 43.63 percent for the long and short queries. These results are consistent with those of the log-based query expansion.

In summary, phrases are very important for searching with short queries. In addition, our method of phrase extraction from user logs, although simple, proved to be effective.

### 6.3 Impact of Number of Expansion Terms

In general, the number of expansion terms should be within a reasonable range in order to produce consistently good performance. Too many expansion terms not only consume

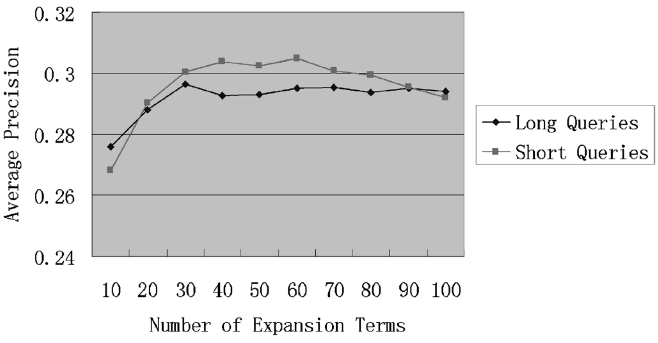

Fig. 3. Impact of number of expansion terms.

more time for the retrieval process, but also have side-effects on the retrieval performance.

We examine the performance of the log-based query expansion by using 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 expansion terms on the two query sets. The results are shown in Fig. 3.

The best performances are obtained with around 30 expansion terms for both query sets. It is worth noting that the curve produced on the long query set is flatter than the other one. The curve of the long query set reaches its summit at 30 expansion terms and remains very flat after 30. In comparison, the curve of the short query set drops after adding more than 60 expansion terms. We attribute this to the fact that the short queries have less original terms, which, when expanded excessively without other terms to serve as context, may produce more side-effects and generate more irrelevant terms. For long queries, as more terms act together in the selection of expansion terms, the chance of generating many irrelevant terms is much less.

## 7 Conclusions

The proliferation of the World Wide Web prompts the wide application of search engines. However, short queries and the incompatibility between the terms in user queries and documents strongly affect the performance of the existing search engines. Many automatic query expansion techniques have been proposed, which can solve the short query and the term mismatching problem to some extent. However, they do not take advantage of the user logs available in various Web sites, and use them as a means for query expansion.

In this article, we presented a novel method for automatic query expansion based on user logs. This method aims first to establish correlations between query terms and document terms by exploiting the user logs. These relationships are then used for query expansion. We have shown that this is an effective way to narrow the gap between the query space and the document space. For new queries, high-quality expansion terms can be selected from the document space on the basis of the extracted correlations. We tested this method on a data set that is similar to the real Web environment. A series of experiments conducted on both long queries and short queries showed that the log-based query expansion method can achieve substantial improvements in performance. It also outperforms local context analysis, which is one of the most effective query expansion methods in the past. Our experiments also show that query expansion is more effective for short queries than for long queries.

## References

[1] M.J. Bates, "Search Techniques." *Ann. Rev. of Information Science and Technology,* M.E. Williams, ed., pp. 139-169, 1981.

[2] D. Beeferman and A. Berger, "Agglomerative Clustering of a Search Engine Query Log," *Proc. SIGKDD,* pp. 407-416, 2000.

[3] G. Brajnik, S. Mizzaro, and C. Tasso, "Evaluating User Interfaces to Information Retrieval Systems: A Case Study on User Support," *Proc. 19th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'96),* pp. 128-136, Aug. 1996.

[4] C. Buckley, G. Salton, and J. Allan, "Automatic Retrieval with Locality Information Using Smart," *Proc. First Text Retrieval Conf. (TREC-1),* pp. 59-72, 1992.

[5] C. Buckley, M. Mitra, J. Walz, and C. Cardie, "Using Clustering and Superconcepts within Smart," *Proc. Sixth Text Retrieval Conf. (TREC-6),* E. Voorhees, ed., pp. 107-124, 1998.

[6] C. Buckley, G. Salton, J. Allan, and A. Singhal, "Automatic Query Expansion Using SMART," *Overview of the Third Retrieval Conf. (TREC-3),* pp. 69-80, Nov. 1994.

[7] C. Carpineto, G. Romano, and B. Bigi, "An Information-Theoretic Approach to Automatic Query Expansion," *ACM Trans. Information Systems,* vol. 19, no. 1, pp. 1-27, Jan. 2001.

[8] J.W. Cooper and R.J. Byrd, "Lexical Navigation: Visually Prompted Query Expansion and Refinement," *Proc. Second ACM Int'l Conf. Digital Libraries,* pp. 237-246, 1997.

[9] W.B. Croft, R. Cook, and D. Wilder, "Providing Government Information on the Internet: Experiences with THOMAS," *Proc. Second Int'l Conf. Theory and Practice of Digital Libraries,* pp. 19-24, 1995.

[10] C.J. Crouch and B. Yang, "Experiments in Automatic Statistical Thesaurus Construction," *Proc. ACM-SIGIR Conf. Research and Development in Information Retrieval,* pp. 77-88, 1992.

[11] H. Cui, J.-R. Wen, J.-Y. Nie, and W.-Y. Ma, "Probabilistic Query Expansion Using User Logs," *Proc. 11th World Wide Web Conf.,* pp. 325-332, 2002.

[12] S. Deerwster, S.T. Dumai, G.W. Furnas, T.K. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *J. Am. Soc. Information Science and Technology,* vol. 41, no. 6, pp. 391-407, 1990.

[13] E. Efthimiadis and P. Biron, "UCLA-Okapi at TREC-2: Query Expansion Experiments," *Proc. Second Text Retrieval Conf. (TREC-2),* D.K. Harmon, ed., 1994.

[14] D. Evans and R. Lefferts, "Design and Evaluation of the CLARIT-TREC-2 System," *Proc. Second Text Retrieval Conf. (TREC-2),* 1994.

[15] G.W. Furnas, T.K. Landauer, L.M. Gomez, and S.T. Dumais, "THE Vocabulary Problem in Human-System Communication," *Comm. ACM,* vol. 30, no. 11, pp. 964-971, 1987.

[16] G. Grefenstette, *Explorations in Automatic Thesaurus Discovery.* Kluwer Academic Publishers, 1994.

[17] S.P. Harter, *Online Information Retrieval: Concepts, Principles, and Techniques.* Orlando, Fla.: Academic Press, 1986.

[18] D. Hull, "Using Statistical Testing in the Evaluation of Retrieval Experiments," *Proc. ACM SIGIR,* pp. 329-338, June 1993.

[19] Y. Jing and W.B. Croft, "An Association Thesaurus for Information Retrieval," *Proc. RIAO,* pp. 146-160, 1994.

[20] M.E. Lesk, "Word-Word Associations In Document Retrieval Systems," *Am. Documentation,* vol. 20, no. 1, pp. 27-38, 1969.

[21] A. Lu, M. Ayoub, and J. Dong, "Ad Hoc Experiments Using EUREKA," *Proc. Text Retrieval Conf. (TREC-5),* pp. 229-240, 1997.

[22] G. Miller, "Wordnet: An Online Lexical Database," *Int'l J. Lexicography,* vol. 3, no. 4, 1990.

[23] M. Mitra, A. Singhal, and C. Buckley, "Improving Automatic Query Expansion," *Proc. 21st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 206-214, 1998.

[24] Y. Qiu and H. Frei, "Concept Based Query Expansion," *Proc. 16th Int'l ACM SIGIR Conf. R & D in Information Retrieval,* pp. 160-169, 1993.

[25] S.E. Robertson and K. Sparck Jones, "Relevance Weighting of Search Terms," *J. Am. Soc. for Information Sciences,* vol. 27, no. 3, pp. 129-146, 1976.

[26] S.E. Robertson, S. Walker, and M. Sparck Jones, et al., "Okapi at TREC-3," *Proc. Second Text Retrieval Conf. (TREC-3),* 1995.

[27] J. Rocchio, "Relevance Feedback in Information Retrieval," *The Smart Retrieval System—Experiments in Automatic Document Processing,* G. Salton, ed., pp. 313-323, 1971.

[28] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval.* England: Pearson Education Limited, 1999.

[29] T. Sakai, S.E. Robertson, and S. Walker, "Flexible Pseudo-Relevance Feedback Via Direct Mapping and Categorization of Search Requests," *Proc. BCS-IRSG ECIR,* pp. 3-14, 2001.

[30] G. Salton, *The SMART Retrieval System—Experiments in Automatic Document Processing.* Englewood Cliffs, N.J.: Prentice Hall, 1971.

[31] G. Salton and C. Buckley, "Improving Retrieval Performance by Relevance Feedback," *J. Am. Soc. for Information Science,* vol. 41, no. 4, pp. 288-297, 1990.

[32] K. Sparck Jones, *Automatic Keyword Classification for Information Retrieval.* London: Butterworths, 1971.

[33] E.M. Voorhees, "Query Expansion Using Lexical-Semantic Relations," *Proc. 17th Int'l Conf. Research and Development in Information Retrieval,* pp. 61-69, 1994.

[34] J.-R. Wen, J.-Y. Nie, and H.-J. Zhang, "Query Clustering Using User Logs," *ACM Trans. Information Systems,* vol. 20, no. 1, pp. 59-81, 2002.

[35] S.K. Wong and W. Ziarko et al., "On Modeling of Information Retrieval Concepts in Vector Spaces," *ACM Trans. Database Systems,* vol. 12, no. 2, pp. 299-321, June 1987.

[36] S.K.M. Wong and Y.Y. Yao, "A Probabilistic Method for Computing Term-by-Term Relationships," *J. Am. Soc. for Information Science,* vol. 44, no. 8, pp. 431-439, 1993.

[37] J. Xu and W.B. Croft, "Query Expansion Using Local and Global Document Analysis," *Proc. 19th Int'l Conf. Research and Development in Information Retrieval,* pp. 4-11, 1996.

[38] J. Xu and W.B. Croft, "Improving the Effectiveness of Information Retrieval with Local Context Analysis," *ACM Trans. Information Systems,* vol. 18, no. 1, pp. 79-112, Jan. 2000.

**Hang Cui** received the BS and MS degrees in management information systems from Tianjin University, Tianjin, China, in 2000 and 2002, respectively. In July 2002, he was admitted into the National University of Singapore, where he is pursuing a PhD degree. In 2001 and 2002, he spent one year working as a visiting student at Microsoft Research Asia, Beijing, China. His research interests include text mining, intelligent information retrieval, machine learning, and Q & A systems.

**Ji-Rong Wen** received the BS and MS degrees in 1994 and 1996, both from School of Information, Renmin University of China. He received the PhD degree in 1999 from the Institute of Computing Technology, the Chinese Academy of Science. He joined Microsoft Research Asia in July 1999 and is currenly a researcher in the Media Management Group. His main research interests are data management, intelligent information retrieval, and Web mining.

**Jian-Yun Nie** received the PhD degree in 1990 from the Université Joseph Fourier of Grenoble, France. He is an associate professor in Département d'informatique et Recherché Opérationnelle, Université de Montréal. His research interests are focused on information retrieval (IR), in particular, cross-language and multilingual IR, knowledge- and NLP-based IR, as well as theoretical aspects of IR such as logical models of IR. He is also interested in data mining.

**Wei-Ying Ma** received the BS degree in electrical engineering from the National Tsing Hua University in Taiwan in 1990, and the MS and PhD degrees in electrical and computer engineering from the University of California at Santa Barbara in 1994 and 1997, respectively. He joined Microsoft Research Asia in April 2001 as the research manager of the Media Management Group. Prior to joining Microsoft, he was with Hewlett-Packard Laboratories in Palo Alto, California, where he was a researcher in the Internet Mobile and Systems Lab. From 1994 to 1997, he was engaged in the Alexandria Digital Library (ADL) project at the University of California at Santa Barbara while completing his PhD degree. Dr. Ma serves as an associate editor for the *Journal of Multimedia Tools and Applications* published by Kluwer Academic Publishers. He has served on the organizing and program committees of many international conferences and has published four book chapters. His research interests include image and video analysis, content-based image and video search and retrieval, machine learning techniques, intelligent information systems, adaptive content delivery, content distribution and services networks, and media delivery and caching. He is a member of the IEEE.