World Wide Web (2010) 13:121–142
DOI 10.1007/s11280-009-0077-1

Finding Related Search Engine Queries by Web Community Based Query Enrichment

Lin Li · Shingo Otsuka · Masaru Kitsuregawa

Received: 28 February 2009 / Revised: 20 July 2009 /
Accepted: 4 November 2009 / Published online: 17 November 2009

Abstract The conventional approaches of finding related search engine queries rely on the common terms shared by two queries to measure their relatedness. However, search engine queries are usually short and the term overlap between two queries is very small. Using query terms as a feature space cannot accurately estimate relatedness. Alternative feature spaces are needed to enrich the term based search queries. In this paper, given a search query, first we extract the Web pages accessed by users from Japanese Web access logs which store the users individual and collective behavior. From these accessed Web pages we usually can get two kinds of feature spaces, i.e, content-sensitive (e.g., nouns) and content-ignorant (e.g., URLs), to enrich the expressions of search queries. Then, the relatedness between search queries can be estimated on their enriched expressions. Our experimental results show that the URL feature space produces much lower precision scores than

The work was done when Lin Li was a Ph.D. student at the University of Tokyo and now she works at Wuhan University of Technology.

L. Li (✉)
Department of Information and Communication Engineering, The University of Tokyo, Tokyo, Japan
e-mail: lilin@tkl.iis.u-tokyo.ac.jp, cathylilin@gmail.com

L. Li
School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China
e-mail: cathylilin@whut.edu.cn

S. Otsuka
National Institute for Materials Science, Tsukuba, Japan
e-mail: otsuka.shingo@nims.go.jp

M. Kitsuregawa
Institute of Industrial Science, The University of Tokyo, Tokyo, Japan
e-mail: kitsure@tkl.iis.u-tokyo.ac.jp

the noun feature space which, however, is not applicable in non-text pages, dynamic pages and so on. It is crucial to improve the quality of the URL (content-ignorant) feature space since it is generally available in all types of Web pages. We propose a novel content-ignorant feature space, called Web community which is created from a Japanese Web page archive by exploiting link analysis. Experimental results show that the proposed Web community feature space generates much better results than the URL feature space.

Keywords relatedness • query enrichment • web access logs • web page archive • web community

1 Introduction

Many advanced searching techniques have been developed and used in commercial Web search engines, such as Google and Yahoo!. Given a term based search query, search engines primarily rely on the matching of the query terms to the document terms in the desired documents to determine which Web pages will be returned. A main problem, however, occurs for search engine users is that if a user is not familiar with some domain, she might fail to choose terms at the appropriate level of representation for her information need, thus having difficulty in organizing and formulating her search query. In this case, recommending related search queries is considered as an effective assistant to help search engine users get their desired Web pages.

Currently, some search engines give suggestions on input queries, thus assisting search engine users in rephrasing their query formulation to improve search quality. *Related search* in Google presents users a bunch of semantically and/or thematically related search terms at the base of a SERP(Search Engine Results Page). In the search box of Yahoo!, *Search Assist* compares an input query to search all others that Yahoo! users have composed and offers suggestions in real time. These services supplied by Google and Yahoo! highlight the importance of finding related search queries. It is intuitive to compute the relatedness between two search queries based on their shared common terms. From a study of a popular search engine logs, Jansen et al. [16] reported that most search queries are short, about two terms per query, the very small term overlap between search queries cannot accurately estimate their relatedness. For example, *New York Times* and *New York subway* have two terms in common, but their retrieved search results are quite different. Moreover, it is possible that queries can be phrased differently with different terms but for the similar information needs. For example, *New York Stock Exchange* has no common terms with *Manhattan*, but *Manhattan* is the largest central business district in the United States and the site of New York stock Exchange. Given this problem, the technique to find semantically related queries (though probably dissimilar in their terms) is becoming an increasingly important research topic that attracts considerable attention.

Query enrichment is an effective method to find semantically related queries, which enriches the representation of a query by alternative feature spaces, instead of using terms in the query itself. Consequently, how to get suitable feature spaces becomes a key in this method. In this paper, given a search query, first we extract the Web pages accessed by users from Japanese Web access logs which store the users individual and collective behavior. From these accessed Web pages we usually can get two kinds of feature spaces, i.e, content-sensitive (e.g., nouns) and content-ignorant (e.g., URLs) which can be used to enrich the expressions of search queries. Then, the relatedness between search queries can be estimated on their enriched expressions, e.g., the overlap of their feature spaces. Our experimental results show that the URL feature space produces lower precision scores than the noun feature space which, however, is not applicable, at least in principle, in settings including: non-text pages like multimedia (image) files, Usenet archives, sites with registration requirement, and dynamic pages returned in response to a submitted query and so forth. It is crucial to improve the quality of the URL (content-ignorant) feature space since it is generally available in all types of Web pages. The problem of the URL feature space is that even though two queries share no common URLs in their accessed Web pages, they may be related since Web pages with different URLs may semantically related.

We are inspired to find a novel content-ignorant feature space for query enrichment. The whole Web can be considered as a graph where nodes are Web pages and edges are hyperlinks. Recent research on link analysis has shown the existence of numerous Web communities[1] on the Web [5, 11–13, 15, 17, 19]. In this context, a Web community is a collection of Web pages with different URLs, but sharing common interest on a specific topic. Our idea is that a query can be enriched by the Web communities that the respective accessed Web pages belong to, instead of using the URLs of Web pages directly. We create Web communities from a Japanese Web page archive by only exploiting link analysis(the technical details will be described in Section 2.2), thus they are regarded an alternative content-ignorant feature space. The proposed Web community feature space is novel, different from the traditional URL and noun feature spaces which are widely used in the literature [1, 2, 6, 8, 33, 34]. Experimental results show that the novel Web community feature space generates much better results than the traditional URL feature space. We also empirically analyze and compare the performance of the three feature spaces (i.e., URL, Web community, and noun) in a query recommendation system which suggests a list of related search queries given an initial input query. Users can utilize the suggested related search queries to tune or redirect the search process. We also reveal that different feature spaces of query enrichment show different characteristics in finding related search queries.

The rest of this paper is organized as follows. Firstly, we introduce how to get the Web community feature space for query enrichment and the relatedness definition between two enriched queries in Section 2. Then, we address the details of experiment methodology in Section 3. Experimental results are discussed in Section 4. Lastly, we review related work and conclude our work in Sections 5 and 6 respectively.

[1] In the field of social network, Web community is also used to mean a set of users having similar interests, which slightly differs from the definition in this paper.

2 Relatedness definition based on query enrichment

Our goal is to find the related search queries given a current query input by a search engine user. Consequently, we need to measure the relatedness between queries and then recommend the top ranked queries. We will first discuss that what kind of feature spaces can be used for query enrichment and how to get them, and then we will give the definition of relatedness based on these feature spaces.

2.1 Discussions

It is not solid to estimate the relatedness between two queries based on their term overlap. Usually, query-result-vectors present a better similarity metric than query term-vectors [26]. Our work follows the recent direction of using log data [1, 2, 8, 33]. It means that we augment a query by the feature spaces extracted from the Web pages accessed by search engine users. This log data based method utilizes the search history of users and is two-fold. On the one hand, the access information as implicit relevance can be automatically obtained. While collecting explicit relevance is labor-intensive since users have to manually give their judgments. On the other hand, one important assumption behind this log data based method is that all the accessed Web pages are *relevant* to the query, which is not as accurate as explicit relevance judgment in the traditional relevance feedback. It is true that choices made by a small number of users are likely to be unreliable. The large amount of information available in our Web access logs makes this less of a problem; we assume that users are more consistent in their choices of relevant Web pages than irrelevant ones. It is therefore reasonable to regard the accessed Web pages as relevant examples from a statistical viewpoint.

Given a query, we extract feature spaces from the accessed Web pages of our Japanese Web access logs. Each query is represented by its respective feature space. For example, if the feature space is the URL of a Web page, a query is enriched or represented by the URLs of all the accessed Web pages; if the feature space is the terms in a Web pages, a query is represented by the terms of all the accessed Web pages. Then, the relatedness between two queries can be estimated based on the similarity of their feature spaces, instead of their term overlap. The URLs of Web pages are content-ignorant and commonly available features, but the URL based query enrichment shows low precision in our experiments where the number of relevant queries is only a little more than the number of irrelevant queries. In addition, it is intuitive to use the contents of Web pages for query enrichment. However, content-sensitive based feature space (e.g., nouns) is not applicable in some cases as we discussed in Section 1. Therefore, we are motivated to improve the precision of finding related search queries using the URL (content-ignorant) based query enrichment. Next, we will introduce a novel content ignorant feature space, called Web community and how to create it using only linkage information between Web pages.

2.2 Creating web communities as content-ignorant feature space

Directly using URLs as a feature space for query enrichment is simple but not very effective. It is understandable that even though the URLs of two Web pages are different, their contents may be related. We think if we can cluster related Web pages into a Web community which is a collection of Web pages sharing common interest on a specific topic, each query will be represented by the Web communities which the accessed Web pages belong to. Our experimental results show that the Web community based query enrichment largely improves the precision of finding related queries over the URL based one.

We create Web communities using only linkage information between pages without taking into account the contents of Web pages. This means that we need a Web page archive which stores the hyperlink information among Web pages. These Web pages are not only the accessed Web pages in our access logs, but also other Web pages on the Web during the period of collecting our logs. Thus, we can cluster Web pages using their linkage information. A large-scale snapshot of a Japanese Web page archive we used was built in February 2002. We crawled 4.5 million Web pages. A connectivity database is built to search outgoing and incoming links of a given Web page in the archive. To create Web communities, we used a manually maintained page list as a seed set which includes 4∼691 pages of companies, organizations and schools. And we extended the seed set by applying *Companion–* which was described in [31].

Here we briefly describe the *Companion–* algorithm that is an important step in our Web community extraction. First, we build a vicinity graph, which is a subgraph of the web around a seed. A vicinity graph is a directed graph, (V, E), where nodes in V represent Web pages, and edges in E represent links between these pages. The vicinity graph includes nodes that can be reached from the seed page by following incoming links then outgoing links (back-forward set), as illustrated in Figure 1.

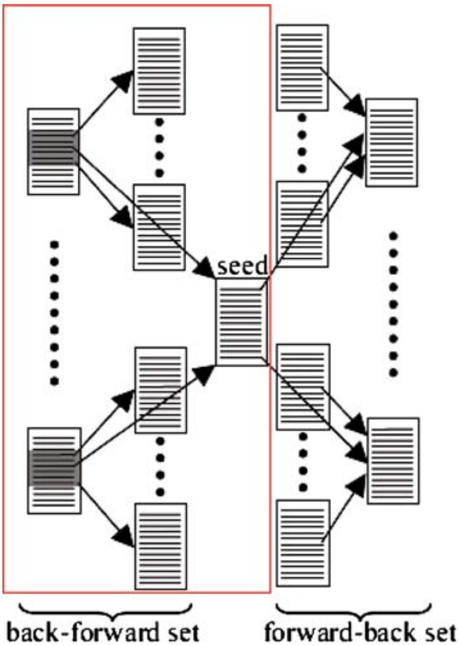

Figure 1 Vicinity graph for our *Companion–* algorithm.

When following outgoing links from each node pointing to the seed in the back-forward set, not all the links are followed but only links (R) immediately preceding the link pointing to the seed, and links (R) immediately succeeding the link. If a node has more than a number of incoming links (Nb), Nb links are randomly selected. In the right part of Figure 1, we also illustrate forward-back set which are built by following outgoing links then incoming links. The famous *Companion* proposed in [9] utilized both back-forward set and forward-back set. Experimental results show that our *Companion*− algorithm using only back-forward set can produce higher precision than *Companion* algorithm by 49.2%. We chose R to be 10, and Nb to be 2000. More result discussions are in [31].

After building vicinity graph, weights are assigned to edges. For each edge, we consider two kinds of weights, an authority weight and a hub weight for decreasing the influence of a single Web page. The authority weight is used for calculating an authority score of each node, and the hub weight is used for calculating a hub score of each node. The notions of authority and hub are proposed by [18]. Simply speaking, an authority is a page with good contents on a topic, is pointed to by many good hub pages. A hub is a page with a list of hyperlinks to valuable pages on the topic, that is, points to many good authorities. We use the following weighting method.

1. If two nodes of an edge have the same server part in their URLs, the weight of the edge has the value 0.
2. If one node has n incoming edges from nodes in the same server, we assign each edge an authority weight of $1/n$.
3. If one node has m outgoing edges to nodes in the same server, we assign each edge a hub weight of $1/m$.

Then we calculate a hub score, $hub(n)$, and an authority score, $auth(n)$ for each node n in the vicinity graph(V,E). The process of the calculation is described in Table 1, where $auth\_weight(n,m)$ and $hub\_weight(n,m)$ represent the authority weight and the hub weight of the edge from n to m, respectively.

Our *Companion*− algorithm computes authority scores for the Web pages in the seed set. Figure 2 illustrates a typical graph of authorities and hubs. In the right part of Figure 2, there are famous computer manufactures which are regarded as authorities. These authorities are densely linked by the hubs in the left part of Figure 2. For each seed, we select the top N authorities, and aggregate them into an extended seed set. We again apply the *Companion*− algorithm to each page in the extended seed set and build a new directed graph where an edge from a node s to

Table 1 Calculation of hub and authority scores.

1:	Initialize $hub(n)$ and $auth(n)$ of each node n to 1.
2:	Repeat the following calculation until $hub(n)$ and $auth(n)$ have converged for each node n.
	For all node n in V,
	$hub(n) \leftarrow \sum_{(n,m)E} \in auth(m) * hub\_weight(n,m)$
	For all node n in V,
	$auth(n) \leftarrow \sum_{(n,m)E} \in hub(m) * auth\_weight(n,m)$
	Normalize $hub(n)$, so that the sum of squares to be 1.
	Normalize $auth(n)$, so that the sum of squares to be 1.
3:	Choose nodes with the N highest authority scores as results.

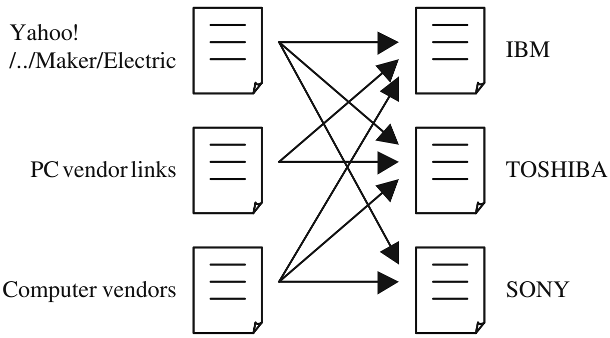

Figure 2 A typical graph of authorities and hubs.

an another node t exists when s derives t as one of the top N authorities. This directed graph is called the authority derivation graph (ADG). ADG built from our seed set includes 13,166 nodes and 70,201 edges.

The next step is to extract a symmetric derivation graph (SDG) from ADG. In this step, we put focus on the symmetric derivation relationships between nodes in ADG, that is, two nodes derive each other using *Companion*−. Using these derivation relationships between pages, we classify pages into communities based on complete bipartite graphs. Finally we automatically create 17 hundred thousand Web communities from one million selected pages. More technical details of creating Web community are in [31].

A part of our community chart is illustrated in Figure 3 where the number represents the community ID. In the center of Figure 3 there is a major community of computer manufactures (15) surrounded by software (18), peripheral device (33 and 16), and digital camera communities (21). It is understandable that even if there are no common URLs of Web pages shared by two enriched queries, they may be related to some extent provided that some Web pages are clustered into a same Web community, e.g., the different Web pages in the computer manufactures (15) community. Each Web page in our data set is labelled by a community ID. Given the accessed Web pages of a query, their corresponding community IDs compose the Web community feature space of the query.

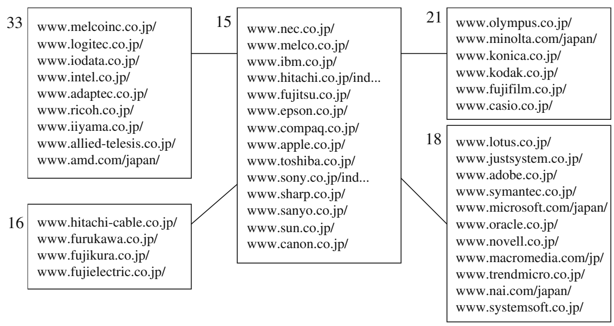

Figure 3 A part of our Web community chart.

2.3 Content-sensitive feature space

We also study a content-sensitive feature space for query enrichment which can be a supplementary information to further enhance performance of finding related queries besides content-ignorant feature spaces (e.g., URL and Web community). As we know, nouns in a document can more accurately represent the topic described by the document than others. Therefore, a query is enriched by the nouns extracted from the contents of its accessed Web page sets. Our noun feature space is created using ChaSen, a Japanese morphological analyzer.[2] We have stored a Japanese Web page archive and completed the morphological analysis of all the Web pages in advance.

2.4 Definition of relatedness

Here we discuss how to calculate relatedness between two queries enriched by a feature space. The goal of this paper is to compare two content-ignorant feature spaces, i.e., URL and Web community, and also empirically study the performance of the noun feature space. Optimally, we had better design different similarity measures for different feature spaces in order to achieve the best performance for each feature space. Since we want to give empirical evidence as to how different feature spaces affect the quality of finding related queries, we need a similarity measure which can be easily applied to all the three feature spaces so that the experimental results based on this measure are reliable and fair.

Let $Q = \{q_1, q_2, \ldots, q_i, \ldots, q_n\}$ be a universal set of all search queries where n is the number of total search queries. Given the accessed Web pages of the query q_i, we extract three feature spaces: 1) URL: $U_{q_i} = \{url_1, url_2, \ldots, url_k\}$, 2) Web community: $C_{q_i} = \{com_1, com_2, \ldots, com_l\}$, and 3) Noun term: $T_{q_i} = \{t_1, t_2, \ldots, t_m\}$. The query q_i is enriched by one of three feature spaces, and then the relatedness between two queries can be estimated on the element intersection of their feature spaces. The popular Jaccard measure is intuitively suitable in this case, defined as

$$Jaccard(q_i, q_j) = \frac{|F_{q_i} \cap F_{q_j}|}{|F_{q_i} \cup F_{q_j}|}, \tag{1}$$

Here F_{q_i} denotes one of the three feature spaces, U_{q_i}, C_{q_i}, and T_{q_i}. Jaccard measure considers the number of common elements shared by two enriched queries.

From our log data, we can get the submit times of each query by various users, and each element in a feature space has its own occurrences, e.g., noun term frequency. Therefore, we are motivated to enhance the Jaccard measure by taking into account the frequency information. For each element in the feature space of a query, we assign it a weight which is the value of its occurrences in this feature space multiplied by the submit times of the query by various users. Then, for normalization, the calculated weight is divided by the sum of the weights of all the elements in the feature space of the query. Finally, we use the sum of the normalized weights of the common

[2] http://chasen-legacy.sourceforge.jp/

elements shared by two queries to estimate their relatedness. The normalized weight of each element in the feature space of a query considers the element frequency (occurrences) because more frequent elements are more likely to be better indicators for a query. Moreover, it also includes the popularity of the query (submit times by various users). Using this kind of normalized weight for relatedness computation means that we can find not only related queries based the quality of their feature spaces, but also popular queries favored by the majority of Web users.

We give a concrete example to understand our idea clearly. For the two search queries “bank” and “deposit”, using the Web community feature space, we get their enriched representations as follow (number denotes Web community ID):

$$q_{bank} = \{37652{,}7968{,}1105{,}38251{,}9650{,}118781{,}140963{,}57665{,}150439{,}35392{,}37750{,}47811\},$$
$$q_{deposit} = \{9650{,}140963{,}40786{,}38251{,}46820{,}37652{,}35392{,}1105\}.$$

Based on our log data, the sum of the weights of all elements in the community space of “bank” is 113 and the sum of “deposit” is 13. Then, the normalized weights of these elements are

$$C_{q_{bank}} = \left\{\frac{7}{113},\frac{3}{113},\frac{3}{113},\frac{2}{113},\frac{2}{113},\frac{1}{113},\frac{1}{113},\frac{1}{113},\frac{1}{113},\frac{1}{113},\frac{1}{113},\frac{1}{113}\right\},$$
$$C_{q_{deposit}} = \left\{\frac{2}{13},\frac{2}{13},\frac{1}{13},\frac{1}{13},\frac{1}{13},\frac{1}{13},\frac{1}{13},\frac{1}{13}\right\}.$$

For example, in $\frac{7}{113}$, 7 is the value of the occurrences of the Web community 37652 multiplied by the submit times of the query *bank* and 113 is the sum of weights of all the communities in $C_{q_{Bank}}$. The intersection of the two enriched queries includes six communities listed as follows:

$$C_{q_{bank}} \cap C_{q_{deposit}} = \{37652{,}1105{,}38251{,}9650{,}140963{,}35392\}.$$

Therefore, the normalized weighted Jaccard score of the two queries, “bank” and “deposit” is: $\frac{0.142 + 0.615}{2} = 0.378$.

Notice that in the above example we only list the communities of $C_{q_{bank}}$ ($C_{q_{deposit}}$) which are not in a so-called excluded set. The reason is that the excluded set stores the *elements with high query frequency* in the three feature spaces. For example, the highly frequent elements of URLs are *Yahoo!, MSN, Google* and so on, and the highly frequent elements of nouns are *I, today, news* and so on. An element in a feature space is in the excluded set if the number of the test queries which feature spaces include the element are more than half of the number of all the test queries used in our evaluation. An element with high query frequency might be less informative to represent a distinct query.

2.5 Other available relatedness measures

We would like to note that our work can easily incorporate other similarity and specificity measures. For example, L_1, Ward and Jensen-Shannon (JS) divergence [21] are common to measure the distance (similarity) between distributions. In text processing, a natural measure of similarity of two documents is the similarity between their word conditional distributions. Roughly speaking, documents with similar conditional word distributions would be related. This idea was first introduced in [25] and was called "distributional clustering". Similarly, in our problem we can recommend related queries with similar conditional distributions of features in each feature space. Another available similarity function is the well-known cosine coefficient which is widely employed in information retrieval [22]. The cosine coefficient is a measure of similarity between two vectors of n dimensions by finding the cosine of the angle between them.

Due to the fact that this paper focuses on an empirical study of different *feature spaces* for query enrichment to improve the quality of finding related queries, we omit the further discussion on more complex relatedness measures. In addition, experimental results show that our relatedness definition can effectively find related search queries.

3 Experiment methodology

We evaluate the effectiveness of our proposed Web community feature space, the URL and noun feature spaces and discuss the characteristics of the three feature spaces. In addition, because the techniques of query recommendation used in commercial search engines are usually confidential, it is difficult to do a quantitative evaluation. We design a query recommendation system which let us easily know the differences between our method and Google search engine by analyzing some concrete examples.

3.1 Web access logs

Our Web access logs, also called "*panel logs*" are provided by *Video Research Interactive Inc.* which is one of Internet rating companies. The collecting method and statistics of this panel logs are described in [23]. Here we give a brief description. Panels (users) are randomly selected based on RDD(Random Digit Dialing), and are requested to install a software that automatically reports web access log to the server of Video Research Interactive. The panel logs consist of *user ID, access time of Web page, reference seconds of Web page, URL of accessed Web page* and so on. The data size is 10GB and the number of users is about 10 thousand.

In this study, we need to extract past search queries and related information from the whole panel logs. Figure 4 shows the details of a part of our panel logs. We notice that the URL from a search engine (e.g., Yahoo!) records the query submitted by a user, as shown in Figure 4a. We extract the query from the URL, and then the access logs followed this URL in a session are corresponding Web pages browsed

Figure 4 A part of our panel logs (Web access logs).

UserID	AccessTime	RefSec	URL
1	2002/9/30 00:00:00	4	http://www.tkl.iis.u-tokyo.ac.jp/welcome_j.html
2	2002/9/30 00:00:00	6	http://www.jma.go.jp/JMA_HP/jma/index.html
3	2002/9/30 00:00:00	8	http://www.kantei.go.jp/
4	2002/9/30 00:00:00	15	http://www.google.co.jp/
1	2002/9/30 00:00:04	6	http://www.tkl.iis.u-tokyo.ac.jp/Kilab/Welcome.html
5	2002/9/30 00:00:04	3	http://www.yahoo.co.jp/
6	2002/9/30 00:00:05	54	http://weather.crc.co.jp/
2	2002/9/30 00:00:06	11	http://www.data.kishou.go.jp/maiji/
3	2002/9/30 00:00:08	34	http://www.kantei.go.jp/new/kousikiyotei.html
5	2002/9/30 00:00:07	10	http://search.yahoo.co.jp/bin/search?p=%C5%B7%B5%A4
5	2002/9/30 00:00:10	300	http://www.tkl.iis.u-tokyo.ac.jp/Kilab/Members/members-j.html

(a)

by the user. The maximum interval to determine the session boundary is 30 min, a well-known threshold [4], such that continuous accesses within 30 min interval are regarded as in a same session. Finally, we got about 125 thousand Japanese queries and 1 million accessed Japanese Web pages.

To obtain the Web community feature space, the other data set used is a Japaneses Web page archive crawled in February 2002. We already introduced it in Section 2.2 where we discussed how to create the Web community feature space from the page archive. For each Web page in the access logs, we want to find its Web community ID in the archive. The time of Web page crawling for the Web page archive is during the time of the collection of access logs. Thus, there are some Web pages which are not covered by the crawling due to the change and deletion of accessed Web pages. We did a preliminary analysis that the URL overlap between the Web access logs and Web page archive is only 18.8%. After we chopped URLs to their hostnames, the overlap increases to 65%. This means that we find the Web community ID of an accessed Web page if the hostname of the URL of the accessed Web page belongs to a Web community. Moreover, our previous work [24] says that using the full URL path shows unsatisfactory performance and its average precision is lower than the noun feature space by 55.65%. We are now doing experiments to evaluate the effect of different path levels of a URL on experimental results. For example, we are using Web page's URL with one path element removed, two path elements removed and so on until we are left with just a hostname.

3.2 Evaluation method

3.2.1 Evaluation process

For each test query, our approach outputs a recommendation list consisting of the top related queries according to their relatedness scores computed by Eq. 1. In addition, each test query will have three recommendation lists generated from the three feature spaces (i.e., URL, community, and noun). For evaluation, we have to judge whether the recommended queries of each three lists are related to the test query or not.

We invited nine volunteers(users) to give such judgments. They are our lab members who usually use search engines to meet their information needs. Nine test queries used in our evaluation are listed in Table 2. In addition, the relevance judgment from users has five levels, i.e., irrelevant, lowly relevant, relevant, highly relevant, and un-judged. Users chose one from the five relevance levels for each pair of a query and a recommended query.

Evaluation is not an easy job because there is not an objective test data set for our problem at the current stage and we have to collect users' judgments manually.

Table 2 Test queries for evaluation.

Test query	#accessed pages	Group	Test query	#accessed pages	Group
Lottery	891	A	bank	113	C
Ring tone	446	B	fishing	64	A
Movie	226	C	scholarship	56	B
Hot spring	211	A	university	50	C
Soccer	202	B			

Furthermore, we should evaluate the effectiveness of three feature spaces for each query, let alone parameter study. In our query recommendation, each search query will be suggested top 20 related queries. The total number of queries evaluated by users are 540.[3] To alleviate the workload on an individual user, we divided the nine users to three group (i.e., A, B, and C) as shown in Table 2 and asked each group to give their relevance judgments on three queries. In Section 4 we will do two statistical analyses to show the judgment consistency of users and the improvement significance since the number of queries and users is not very large in our evaluation.

3.2.2 Evaluation measure

After obtaining the users' judgments, we can know the judgment levels of recommended queries in each recommendation list. The evaluation results of each feature space are got by computing the percentage of the number of queries judged as one of the five relevance levels over all the recommended queries of a list.[4] In other words, we concern that at each relevance level there are how many the queries of each recommendation list. If we consider the queries which are judged as "relevant" or "highly relevant", the percentage is the *Precision* [22] score which is defined as the percentage of the number of related queries in a recommendation list.

Moreover, MAP and $NDCG$ [22] which are widely used for ranking problems, especially in Web search besides *Precision*. Because the length of a term based query is much shorter than that of a Web page, search engine users can read the top ten or twenty recommended queries much more quickly than they browse and click the top Web pages one by one. In this case, we are interested in the number of related queries in a recommendation list. Therefore, for each feature space we report its percentages at each relevance level as evaluation results.

4 Evaluation results and discussions

4.1 Kappa statistics

After collecting users' judgment results, we study the quality of users. We want to know the variability of user's categorical ratings to measure user disagreement which tell us how users classify individual subjects (queries) into the same category (relevance level) on the measurement scale. The judgments from different users should largely reach a good agreement for a same test query.

Kappa statistics is one of the most common approaches [29]. Kappa can be thought of as the chance-corrected proportional agreement, and possible values range from +1 (perfect agreement) via 0 (no agreement above that expected by chance) to −1 (complete disagreement). Table 3 provides a rough guide of what is a good agreement. We require users to answer questionnaires that supply two recommended

[3] 9 search queries * 20 recommended queries * 3 feature spaces = 540 evaluated queries.

[4] 9 search queries * 20 recommended queries = 180 evaluated queries.

Table 3 Kappa and strength of agreement.

Kappa	Strength of agreement	Kappa	Strength of agreement
0.00	Poor	0.41–0.60	Moderate
0.01–0.20	Slight	0.61–0.80	Substantial
0.21–0.40	Fair	0.81–1.00	Almost perfect

queries per test query.[5] Because two users are grouped as a pair to compute a Kappa value, the total number of test pairs is 36 $\left(C_2^9 = 36\right)$.

We collected the relevance judgment results of the nine test queries and the average of all Kappa values is 0.388. This value is not in the range 0.41–0.6, a moderate agreement in general. As we examined the contents of the questionnaire in detail, we noticed that several users chose the "un-judged" level in the recommended queries while most other users gave them a same evaluation (e.g., highly relevant). It is the reason that results in the fall of the Kappa value. Therefore, we deleted recommended queries which are evaluated as "un-judged" by users in the calculation of Kappa statistics. Then, the average value of our Kappa statistics became 0.41. In addition, there is a difference between the choice frequencies of the "relevant" and "highly relevant" judgments. We calculated the average Kappa value again by treating the two judgments as equivalence. Finally, we got a value of 0.508. To sum up, in terms of the statements in Table 3, the agreement in our results is moderate at 95% confidence level. The number of the users is not very large, but our Kappa statistics analysis shows that the quality of the users is satisfactory. Thus, their judgments are reliable for evaluation.

4.2 T-test for statistical testing

We will compare the performance of the three feature spaces, i.e., URL, community, and noun. Since the number of test queries is nine, a paired t-test for statistical testing is performed to verify whether the improvements of the community and noun feature spaces over the URL feature space are *statistically significant* or not.

First, we combine the percentage values of "relevant" and "highly relevant". Then, we have three samples each of which consists of these combined percentage values of each feature space. The sizes of the three sample are equal, i.e., the number of test queries. Last, the pairwise T-tests are done on the three feature spaces. Table 4 lists the results of t-test statistical analysis with a confidence level of 95%. In Table 4 YES means the mean difference of the percentage values of two feature spaces is significant at the 95% level. Moreover, experimental results also tell us that the largest average percentage value is produced by the noun feature space, the second one is the community feature space, and the smallest value is produced by the URL space. Therefore, we can say that the improvements of the noun and community feature spaces over the URL feature space are statistically significant when the experiments are conducted on nine test queries. In the following, we will give how much percents the improvements are.

[5] 9 search queries * 2 recommended queries = 18 evaluated queries.

Table 4 Paired t-test results for statistically significant testing.

	Community vs. URL	Noun vs. URL	Noun vs. community
Significance	YES	YES	YES

4.3 Comparisons of different feature spaces in query enrichment

The precision scores averaged by all the test queries are shown in Table 5 where we combine the results of highly relevant and relevant judgments. For example, The precision score of the "relevant + highly relevant" in the column "noun space" is 0.843 which means the percentage of the number of queries judged as "relevant" or "highly relevant" over all recommended queries using the noun based query enrichment.

In Table 5, using the URL space, the number of recommended queries judged as "irrelevant" is more than using the community and noun spaces, while the number of recommended queries judged as "relevant + highly relevant" is less than other two spaces. Using the proposed Web community space, the number of recommended queries judged as "relevant + highly relevant" is more than using the URL space and the improvement is about 37.3%. While using the Web community space, the number of recommended queries judged as "irrelevant" is less than using the URL space and the improvement is about 28.45%. This means that our Web community based query enrichment is effective and largely improves the recommendation precision over using URL space.

If the contents of Web pages are accessible, when the noun space based strategy is applied, about 80% of all recommended queries are evaluated as "relevant + highly relevant", while only about 3.7% of all recommended queries are evaluated as "irrelevant", which shows better results than the other two feature spaces. This result is easily understandable. Getting more information from Web pages to enrich a query can help us achieve better performance. As we discussed in Section 1, since there are several kinds of Web pages which are difficult to extract their contents, the content-ignorant feature spaces like URL and Web community is still of the essence. Our experimental results show that the proposed Web community feature space to enrich a query, which largely improve the recommendation precision over the URL feature space, and the noun features space can be a very useful supplementary source to further achieve higher precision.

The evaluation results of each individual test query are presented in Table 6. The community based strategy can give comparative results with the noun based strategy for some queries, e.g., "university", "hot spring", and "bank" queries; only for "lottery", the precision score of the URL based strategy is 0.833 which is close

Table 5 Evaluation results of the recommended queries with four relevance levels.

Relevance	URL	Community	Noun
Irrelevant	0.341	0.244	0.037
Lowly relevant	0.107	0.089	0.043
Relevant + highly relevant	0.445 (0.106+0.339)	0.611 (0.131+0.480)	0.843 (0.135+0.707)
Un-judged	0.107	0.056	0.078

Table 6 Evaluation results of individual test queries.

Query	Relevance	URL	Com	Noun
Lottery	Irrelevant	0.100	0.450	0.100
	Lowly relevant	0.050	0.050	0.033
	Relevant	0.000	0.033	0.033
	Highly relevant	0.833	0.450	0.833
	Relevant + highly relevant	**0.833**	0.483	**0.866**
	Un-judged	0.017	0.017	0.000
Movie	Irrelevant	0.000	0.050	0.017
	Lowly relevant	0.067	0.050	0.067
	Relevant	0.067	0.200	0.183
	Highly relevant	0.550	0.483	0.567
	Relevant + highly relevant	0.617	**0.683**	**0.750**
	Un-judged	0.317	0.217	0.167
Soccer	Irrelevant	0.417	0.000	0.050
	Lowly relevant	0.000	0.017	0.017
	Relevant	0.117	0.050	0.067
	Highly relevant	0.267	0.733	0.817
	Relevant + highly relevant	0.384	**0.783**	**0.884**
	Un-judged	0.200	0.200	0.050
Fishing	Irrelevant	0.767	0.617	0.000
	Lowly relevant	0.050	0.200	0.017
	Relevant	0.100	0.167	0.150
	Highly relevant	0.000	0.000	0.717
	Relevant + highly relevant	0.100	0.167	**0.867**
	Un-judged	0.083	0.017	0.117
University	Irrelevant	0.250	0.100	0.017
	Lowly relevant	0.083	0.133	0.033
	Relevant	0.117	0.017	0.183
	Highly relevant	0.450	0.733	0.733
	Relevant + highly relevant	0.567	**0.840**	**0.916**
	Un-judged	0.100	0.017	0.033
Ring tone	Irrelevant	0.250	0.083	0.000
	Lowly relevant	0.133	0.050	0.017
	Relevant	0.050	0.067	0.067
	Highly relevant	0.567	0.800	0.900
	Relevant + highly relevant	0.617	**0.867**	**0.967**
	Un-judged	0.000	0.000	0.017
Hot spring	Irrelevant	0.100	0.150	0.000
	Lowly relevant	0.133	0.067	0.000
	Relevant	0.383	0.233	0.100
	Highly relevant	0.233	0.550	0.700
	Relevant + highly relevant	0.616	**0.783**	**0.800**
	Un-judged	0.150	0.000	0.200

Table 6 (continued)

Query	Relevance	URL	Com	Noun
Bank	Irrelevant	0.467	0.150	0.050
	Lowly relevant	0.283	0.117	0.167
	Relevant	0.050	0.233	0.233
	Highly relevant	0.117	0.467	0.483
	Relevant + highly relevant	0.167	**0.700**	**0.716**
	Un-judged	0.083	0.033	0.067
Scholarship	Irrelevant	0.717	0.600	0.100
	Lowly relevant	0.167	0.117	0.033
	Relevant	0.067	0.183	0.200
	Highly relevant	0.033	0.100	0.617
	Relevant + highly relevant	0.100	0.283	**0.817**
	Un-judged	0.017	0.000	0.050

to that of the noun based strategy (i.e., 0.866). These results further confirm that the Web community based query enrichment is more effective than the URL based enrich, while noun based enrichment is the best selection, but only in the case that we can get the contents of Web pages.

4.4 Case study using our query recommendation system

To conveniently do case study and show the differences between our results and "Google Suggestion", we designed a query recommendation system which finds related past queries given an input query. Its interface is illustrated in Figure 5. A user can input a search query in Figure 5(1) while the related queries recommended are divided into three parts according to different feature spaces. Then, the user can choose one recommended query to add or replace the initial query and submit the reformulated query to a search engine from a drop list as shown in Figure 5(2). Finally, the search results retrieved by the selected search engine are listed in the right part. Furthermore, in Figure 5(3), there are two slide bars which can adjust the lower and upper bounds of relatedness scores. For each feature space, the maximal number of recommended queries is 20. If the user wants more hints, she can click a button shown in Figure 5(4) to get more recommended queries ordered by their relatedness scores with the initial query. Figure 5 presents the recommendation of the query "bank" as an example. Since in this study we utilize Japanese Web data, the corresponding English translation is in the bottom of this figure. If some queries are only available in Japanese, we give a brief English explanation. For example, the query "Mizuho" is a famous Japanese bank.

A Web community (a collection of Web pages sharing a related topic)includes Web pages with different URLs. The number of common elements of two queries enriched by Web communities may be more than that of URL based strategy. Using the query "bank" in Figure 5 as an example, the community "city banks" has A bank and B bank. Using URL based strategy, the URLs of the homepages of the two banks are different, so they are not included in the common set, while using community based strategy, they belong to a same community, so they are included in the common set. For example, "Mizuho bank", "Suito shinkin bank", and "Yamanashi chuo

Google Suggestion

bank interest
bank comparison
bank ranking
bank business hours
bank grading
bank charge
bank transfer charge
bank code
bank transfer
bank interest comparison

(Noun space)

"major consumer product maker"	"Mizuho bank"	"exchange rate"	"Sony bank"
"unification of the banks"	"Sony"	"foreign currency deposits"	(name of financial commodity)
"Koizumi cabinet inauguration"	"rate" "megabank"	(name of financial commodity)	(name of financial commodity)
"deposit"	"Hitachi"	"Mizuho" (bank name)	"Japanese-English dictionary"
"restructure"	"pay off"	"Yen rate"	"my car loan"

(Community space)

"deposit"	"combative sport"	"Mizuho" (bank name)	"exchange rate"
"Asahi bank"	"national athletic meet"	"my car loan"	"bankruptcy information"
"welfare fixed deposit"	"advanced payment"	"Saltlake" (city name)	"Yamanashi Chuo bank"
"toto"(football lot)	(a name of public lottery)	"break-even point"	"Suito shink in bank"
"Kumiko Endo"	"announcer"	"beauty"	"Dai-I chi Kangyo bank"

(URL space)

"Kumiko Endo"	"national athletic meet"	"Saltlake" (city name)	"traffic jam information"
"Tazawa lake"	(company name)	"marathon"	"cherry tree"
"deposit"	"announcer"	(company name)	"Mizuho" (bank name)
"combative sport"	"traveler's check"	"cellular phone" "bulletin board"	(company name)
"Kawabe"	"cleanness freak"	"wine"	"Syunsuke Nakamura"

Figure 5 The user interface of our system.

bank” are popular banks in Japan. Moreover, our community extraction algorithm keeps the Web pages with high authority scores and deletes the Web pages with low authority scores in a community, which lays a foundation for finding related queries with high quality.

4.5 Differences with Google suggestion

We illustrate the results of our case study in Figures 5, 6, and 7 to show the differences between our system and the query recommendation service provided by *Google Suggestion*. It is difficult to do a quantitative comparison with Google suggestion which has been trained on queries submitted to Google, while our tool has been trained on a different data set. It is not suitable to conclude which method is better, because two main factors play in this comparison: the different training data sets and the methods used which are likely different too. In addition, the suggestion technique of Google is confidential. Therefore, we use some concrete examples as case study to show the differences.

The suggestion results of Google were obtained in 2007 when our experiments were conducted. In Figure 5 our system presents many good related recommended queries concerned with *bank* in the noun and community spaces such as “deposit”, “my car loan”, “toto” and so on that are not given by *Google Suggestion*. There are better recommended queries using noun space than *Google Suggestion* in the example of *Fishing* as shown in Figure 6. For example, “fishing aa” recommended

Search Query "Fishing"

Nouns pace		(F) means "fish name"	
"Shibayama"(F)	(F)	"lure" "Kiya-kyusyu city" "seabass"	"surff ishing" "M ihama word"
"(F) fishing" "Chiba city"	"rig" (F) "fishing"	"fishing" "lakeToyota" "point"	(F) (place name) "Hiroshima"
"January" "seabass" "minnow"	"Hyogo Meichoukai"	"Marukyu" "worm"	"fish guide"
"Zushi city" "point" (F)	"PEN reel" "Direct Marketing"	"rig" "fishing" (F)	"suma marine fishing park"
"Haneda Turimasa"	"fishing" "Yukuhashi city"	"rig" (F)	"Osaka bay" "point"
Community space			
"Kagoshima city"	"sea"	"part-time job"	"music"
"prayer"	"car"	"swimming beach"	"gsx1400"
"job offer"	"mah-jongg"	"job offer information"	"wine"
"christmas card"	"yahoo"	"job"	"house-moving"
"cad"	"chat"	"rental server"	"horse racing"
URL space			
"vector"	"Yasuo Uchiba"	"make links"	"BBS"
"computation" "capacity" "windows 95"	"white day"	"Youichi Sasakura"	"cad"
"bulletin board" "license" "CAD qualification"	"books"	"Light"	"never7" (gametitle)
"printer"	"piano" "midi"	"golf"	"playstation"
"simcity" (game title)	"smoked"	"constellation fortune-telling"	"mac"

The Results of Our System

Google Suggestion
fishing aa
fishing rig
fishing video
fishing game
fishing beginner
fishing blog
fishing weather
fishing how to knot
fishing winter
fishing bar

Figure 6 The results of our system and Google Suggestion for query: "Fishing".

by "Google Suggestion" is not highly relevant to "fishing" while the noun based strategy gives "Marukyu", "worm", "fish guide" and so on which are more related to "fishing". We observe that those suggested queries returned from Google are rather similar in their terms and usually share common term. For instance, if a user

Search Query "Soccer"

Nounspace			
(football team)	"world cup" "image"	"w cup"	"England"
"fifa"	"Batistuta"	"soccer shop"	"Japan national team"
"world cup"	"World cup soccer"	"official shop"	"Inamoto player"
"Ilhan"(football player)	"Japan emperor's cup" "news flash"	"bikini"	"gossip" "Takayuki"
"Nakata Hide"	"soccer world cup"	"Beckham" "photo"	"cup fifa world"
Community space			
"Nabisco cup"	"soccer information"	"Tatsuya Enoki"	"Kazushi Kimura"
"soccer site"	"Masahiro Endo"	"England"	"world cup"
"Danish soccer"	"DF" "soccer"	"Europe" "soccer"	"jawoc"
"Alan Shearer"	"Passarella"	"Japan national team"	"senior soccer"
"2882017.0"	"toto" (football lot)	"f" "marinos"(football team)	"asian cup"
URL space			
"Japan emperor's cup"	"Koizumi cabinet"	"Byrom Inc"	"princess aiko"
"sports news"	"world cup"	"Sarah Hughes"	"Yuko Yamaguchi"
"yahoo! nba"	"woman pole vault" "dragila"	"professional baseball flash"	"soccer world cup"
"toto" (football lot)	"Japan national team"	"Ehime Maru"	"Nekohachi Edoya"
"f1"	"Ikko Tanaka"	"senior soccer"	"christen" "princess aiko"

The Results of Our System

Google Suggestion
soccer transfer
soccer Japan national team
soccer uniform
soccer transfer information
soccer video
soccer spike
soccer news
soccer rule
soccer senior high school
soccer blog

Figure 7 The results of our system and Google Suggestion for query: "Soccer".

searches for *Soccer*[6] in Google Japan, the following related queries are presented: *soccer transfer*, *soccer Japan national team*, *soccer uniform*, *soccer video*, and so on. The recommended queries by our system give some football player such as *Beckham* and *Batistuta*, as shown in Figure 7. Moreover, our panel logs are gathered in 2002 when Koria-Japan World Cup Soccer was held. Our system suggests the related queries of "Soccer" like "cup fifa world" and "world cup image" which do not include the keyword "soccer".

5 Related work

Past search queries embody the collaborative knowledge of users, which can be a useful source for finding the related queries of a current input query. Existing techniques differ from one another in terms of how to get additional feature spaces to enrich query expression.

5.1 Pseudo relevance feedback

Pseudo relevance feedback is widely used. In [10] the authors improved the effectiveness of a user-supplied query by identifying key terms from potentially relevant documents from past queries. In [14], the authors introduced a software agent that collects queries from previous users, and determined the query similarity based on the Web pages returned by queries, and not the actual terms in the queries themselves. In [20], the authors devised a a hierarchical agglomerative clustering (HAC) based rank mechanism to order the related queries using the URLs of returned search results.

5.2 Implicit relevance feedback

On the Web, recent studies [1, 2, 28, 33, 36] are interested in using Web logs as an additional source to enrich short Web queries. This means this kind of methods makes use of the Web pages *clicked* by users, not the whole set of search results returned by a search engine(pseudo relevance feedback). There are two kinds of feature spaces commonly used in the literature, i.e., content-sensitive and content-ignorant features.

Beeferman et al. [2] used single-linkage clustering to cluster related queries based on the common clicked URLs two queries share, a content-ignorant feature space. Wen et al. [33] further proposed three kinds of features to compute query to query relatedness: 1) based on terms of the query, 2) based on common clicked URLs, and 3) based on the distance of the clicked Web pages in a predefined hierarchy. The terms in a short Web query would not give reliable information, while the limitation of URL feature space is that two Web pages with different URLs may be semantically related in contents. The third features in [33] needs a concept taxonomy and requires Web pages to be classified into the taxonomy as well. Such taxonomy is not generally available. Baeza-Yates et al. [1] find related queries based on the

[6]In general, Japanese say *soccer* not *football*.

content of clicked Web pages using click frequency as a weighting scheme. Their experiments show that using the content information of a Web page (e.g., nouns) is a more accurate query enrichment way to measure query similarity than using the URL of a Web page.

Some studies [28, 36] went another direction and they viewed Web logs as a set of transactions where a single submits a sequence of related queries in a time interval. However, it is difficulty to accurately determine transactions of successive queries that belong to the same search process, which is still a open problem now.

Our work aims at improving the quality of URL based query enrichment since the content of a Web page may not always be available. We proposed a novel content-ignorant feature space, i.e., Web community and our experiments show that the Web community feature space produces much higher precision scores than the simple and intuitive URL feature space.

5.3 Query expansion

We design a query recommendation system in this paper which suggest related queries to help users refine their original queries, while query expansion is also an alternative to revise users' queries. The conventional research efforts on query expansion are classified into three categorizations according to the information sources they use to find relevant terms for query expansion: 1) corpus-based statistics analysis [32], 2) relevance feedback based recommendation [27], and 3) local context based recommendation [3, 35]. In addition, some researches combined multiple sources of knowledge on term associations [6, 7, 30]. Others used Web logs to bridge the term gap between user-centric query space and author-centric Web page space [8, 37]. Most query expansion techniques suggest terms used extracted from Web pages. However, some terms are difficult to be suggested because of their high document frequencies, e.g., "good", "delicious", and so on. We think that *past search queries* stored in the query logs may be a source of additional evidence. If these terms appeared in some past queries, we can easily suggest them to users by using the whole query. Terms in related queries can also be a effective source of expansion terms. Further experiments on query expansion using related queries would be an interesting topic in our future work.

6 Conclusions and future work

In spite of the improvement of searching accuracy with the development of technologies, it is not always true that search engine users can hit upon the proper search queries. In this paper, we studied the problem of how to find related search queries which can help users refine their original queries and get their desired Web pages. To enrich the expressions of search queries, directly using the URL feature space is simple but not very effective. It is understandable that even though the URLs of two Web pages are different, their contents may be related. We proposed a novel feature space called Web community. Topic-related Web pages are clustered into a Web community and each query is represented by the Web communities which its accessed Web pages belong to. The relatedness between queries based on the common Web communities they share produces much higher precision than the traditional URL feature space. In addition, although the noun feature space can find more related search queries than the URL and Web community spaces, it is not generally available in the case of non-text pages. Therefore, the noun feature space can be used as a supplementary to further enhance the performance. We also provided empirical evidence as to how different feature spaces (i.e., noun, URL, and Web community) affect the results of finding related queries by using large-scale Web access logs and Japanese Web page archive. Moreover, a query recommendation system was devised to show the suggestion differences between our results and"Google Suggestion", and conveniently analyzed the differences of the three feature spaces of query enrichment.

Our research continues along several dimensions. We are designing an optimization algorithm for incremental updates of related query data and evaluating other similarity measures. We will evaluate the effect of related queries on the quality of Web information retrieval since the goal of finding related queries tries to help users get their desired information by formulating better search queries.

References

1. Baeza-Yates, R.A., Hurtado, C.A., Mendoza, M.: Improving search engines by query clustering. JASIST **58**(12), 1793–1804 (2007)
2. Beeferman, D., Berger, A.L.: Agglomerative clustering of a search engine query log. In: Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'00), pp. 407–416. Boston, MA, USA (2000)
3. Buckley, C., Salton, G., Allan, J., Singhal, A.: Automatic query expansion using smart. In: Proceedings of Text REtrieval Conference (TREC'03), pp. 69–080. Gaithersburg, Maryland (1994)
4. Catledge, L., Pitkow, J.: Characterizing browsing behaviors on the world-wide web. Comput. Netw. ISDN Syst. **27**(6) (1995)
5. Caverlee, J., Liu, L., Rocco, D.: Discovering interesting relationships among deep web databases: a source-biased approach. World Wide Web **9**(4), 585–622 (2006)
6. Chirita, P.A., Firan, C.S., Nejdl, W.: Personalized query expansion for the web. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'07), pp. 7–14. Amsterdam, The Netherlands (2007)
7. Collins-Thompson, K., Callan, J.: Query expansion using random walk models. In: Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management (CIKM'05), pp. 704–711. Bremen, Germany (2005)
8. Cui, H., Wen, J.R., Nie, J.Y., Ma, W.Y.: Query expansion by mining user logs. IEEE Trans. Knowl. Data Eng. **15**(4), 829–839 (2003)
9. Dean, J., Henzinger, M.R.: Finding related pages in the world wide web. Comput. Networks **31**(11–16), 1467–1479 (1999)
10. Fitzpatrick, L., Dent, M.: Automatic feedback using past queries: social searching? In: Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'97), pp. 306–313. Philadelphia, PA, USA (1997)
11. Flake, G.W., Lawrence, S., Giles, C.L.: Efficient identification of web communities. In: Proceedings of the 6h ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'00), pp. 150–160. Boston, MA, USA (2000)
12. Flake, G.W., Lawrence, S., Giles, C.L., Coetzee, F.: Self-organization and identification of web communities. IEEE Comput. **35**(3), 66–71 (2002)
13. Gibson, D., Kleinberg, J.M., Raghavan, P.: Inferring web communities from link topology. In: Proceedings of the 9th ACM Conference on Hypertext and Hypermedia (HT'98), pp. 225–234. Pittsburgh, PA, USA (1998)
14. Glance, N.S.: Community search assistant. In: Proceedings of the 2001 International Conference on Intelligent User Interfaces (IUI'01), pp. 91–96. Santa Fe, NM, USA (2001)

15. Greco, G., Greco, S., Zumpano, E.: Web communities: models and algorithms. World Wide Web **7**(1), 58–82 (2004)
16. Jansen, B.J., Spink, A., Bateman, J., Saracevic, T.: Real life information retrieval: a study of user queries on the web. SIGIR Forum **32**(1), 5–17 (1998)
17. Khy, S., Ishikawa, Y., Kitagawa, H.: A novelty-based clustering method for on-line documents. World Wide Web **11**(1), 1–37 (2008)
18. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM **46**(5), 604–632 (1999)
19. Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A.: Trawling the web for emerging cyber-communities. Comput. Networks **31**(11–16), 1481–1493 (1999)
20. Li, L., Yang, Z., Liu, L., Kitsuregawa, M.: Query-url bipartite based approach to personalized query recommendation. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence,(AAAI'08), pp. 1189–1194. Chicago, Illinois, USA (2008)
21. Lin, J.: Divergence measures based on the shannon entropy. IEEE Trans. Inf. Theory **37**(1), 145 (1991)
22. Manning, C.D., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
23. Otsuka, S., Toyoda, M., Hirai, J., Kitsuregawa, M.: Extracting user behavior by web communities technology on global web logs. In: Proceedings of 15th International Conference on Database and Expert Systems Applications (DEXA'04), pp. 957–968. Zaragoza, Spain (2004)
24. Otsuka, S., Toyoda, M., Kitsuregawa, M.: A study for related words finding methods using global web access logs. IPSJ Transactions on Databases (TOD) **46**, 82–92 (2005)
25. Pereira, F.C.N., Tishby, N., Lee, L.: Distributional clustering of english words. In: Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL'93), pp. 183–190 (1993)
26. Raghavan, V.V., Sever, H.: On the reuse of past optimal queries. In: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'95), pp. 344–350. Seattle, Washington, USA (1995)
27. Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. JASIS **41**(4), 288–297 (1990)
28. Shi, X., Yang, C.C.: Mining related queries from web search engine query logs using an improved association rule mining model. JASIST **58**(12), 1871–1883 (2007)
29. Siegel, S., Castellan, N.J.: Nonparametric Statistics for the Behavioral Sciences, 2nd ed. McGraw-Hill, New York(1988)
30. Sun, R., Ong, C.H., Chua, T.S.: Mining dependency relations for query expansion in passage retrieval. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06), pp. 382–389. Seattle, Washington, USA (2006)
31. Toyoda, M., Kitsuregawa, M.: Creating a web community chart for navigating related communities. In: Proceedings of the 12th ACM Conference on Hypertext and Hypermedia (HT'01), pp. 103–112. Århus, Denmark (2001)
32. Voorhees, E.M.: Query expansion using lexical-semantic relations. In: Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94), pp. 61–69. Dublin, Ireland (1994)
33. Wen, J.R., Nie, J.Y., Zhang, H.: Query clustering using user logs. ACM Trans. Inf. Syst. **20**(1), 59–81 (2002)
34. Xu, J., Croft, W.B.: Query expansion using local and global document analysis. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96), pp. 4–11. Zurich, Switzerland (1996)
35. Xu, J., Croft, W.B.: Improving the effectiveness of information retrieval with local context analysis. ACM Trans. Inf. Syst. **18**(1), 79–112 (2000)
36. Zhang, Z., Nasraoui, O.: Mining search engine query logs for query recommendation. In: Proceedings of the 15th International Conference on World Wide Web (WWW'06), pp. 1039–1040. Edinburgh, Scotland, UK (2006)
37. Zhu, Y., Gruenwald, L.: Query expansion using web access log files. In: Proceedings of the 16th International Conference on Database and Expert Systems Applications (DEXA'05), pp. 686–695. Copenhagen, Denmark (2005)