International Journal of u-and e-Service, Science and Technology
Vol.7, No.4 (2014), pp.263-276
http://dx.doi.org/10.14257/ijunnesst.2014.7.4.24

Query Suggestion Based on Theme and Context[1]

Lingling Meng[1], Runqing Huang[2] and Junzhong Gu[3]

[1]*Department of Educational Information Technology, East China Normal University, Shanghai, 200062, China*
[2]*Shanghai Municipal People's Government, Shanghai, 200003, China*
[3]*Computer Science and Technology Department, East China Normal University, Shanghai, 200062,*
[1]*llmeng@deit.ecnu.edu.cn,* [2]*runqinghuang@gmail.com,* [3]*China jzgu@ica.stc.sh.cn*

Abstract

Query suggestion has become one of the most fundamental features of search engines. It attempts to suggest a series of similar queries for improving the search effectiveness. The paper proposed a new query suggestion method which is based on themes and context. Different from previous work, it measure similar queries from the level of semantic level and a new similar queries metric is presented. Furthermore, how to choose similar queries for suggestion is discussed and a new method is proposed. In the new query suggestion method, not only theme, but also query context has been taken into considerate. Experiments show that the new query suggestion method significantly outperformed than related works.

Keywords: *query suggestion, similar queries metric, semantic information, them, context*

1. Introduction

With the development of computer and networks, the Internet has given rise to a great deal of information. People who enjoy the information are also confused. How to obtain valuable knowledge from Internet is an urgent problem. More and more people depend on search engine. But now, the use of traditional search engines for information retrieval is still hard to make user satisfied. On the one hand, most users can not articulate their needs, and the query words entered by the user is relatively short, coupled with the presence of polysemy, which make it difficult for search engines to determine the user's query intent. On the other hand, traditional information retrievals only take word forms into considered, ignore the semantic factor. An effective way to solve theses problems is to use query suggestion technology. The paper proposed a new query suggestion method. Different from previous works, the new method measure similar queries from the level of semantic level and a new framework for similar queries metric is presented. Furthermore, how to choose similar queries for suggestion is discussed and a new method is proposed. In the new query suggestion method, not only theme, but also query context has been taken into considerate. Experiments show that the new method significantly outperformed than related works.

The rest of this paper is as follows: in Section 2 related works are presented. A new query suggestion method based on theme and context is proposed in Section 3. Section 4 shows the evaluation of the new method, including experiments, data analyzing, and the achievements.

[1] The work in the paper was supported by Shanghai Industry-University Cooperation Foundation (Grant No. Shanghai CXY-2013-84) and Shanghai Scientific Development Foundation (Grant No.11530700300).

ISSN: 2005-4246 IJUNNEST

Conclusion and future Work is described in Section 5.

2. Related Work

In this section, we review the related background work in the area of query suggestion. Query suggestion can date back to 1990s [1-4]. Rutger University carries out a series of experiments to study man-machine interaction of information retrieval system. The results indicate that: compared to automatic query expansion, users prefer to use query suggestion techniques in information retrieval than query expansion and query suggestion can really help to improve the retrieval effectiveness and save search time[5-7]. Since then, the query suggestion technology gradually becomes a hot topic. Now some methods have been proposed.

Befferman and Berger propose a query clustering technique based on common clicked URLs [8]. It views the user query logs as a bipartite graph, with the vertices on one side corresponding to queries and on the other side to URLs. One can apply an agglomerative clustering algorithm to the graph's vertices to identify related queries and URLs. Bruno M. Fonseca uses association rule to measure the similarity of queries [9].In his research it takes query as item and session as transaction of association rule mining. Ji-Rong and Jian-Yun propose a similar queries clustering algorithm to recommend URLs to frequently asked queries of a search engine [10]. It assumed that: (1) If two queries contain the same or similar terms, they denote the same or similar information needs. (2)Two queries are similar if they lead to the selection of the same or similar document. The function of similar queries is defined by combining both assumptions linearly. In Ji-Min Wang's research, a new method for discovering related queries was presented [11]. First, some statistical characteristics of a candidate query for a given query were extracted from the log files, such as the numbers of different users submitted, the numbers of the candidate query submitted as well as the returned result clicked, the numbers of common terms and common URLs clicked between the candidate query and the given query. Then these candidate queries were ranked with a linear regression model learned from human labeled training data.

All the measures above are simple and effective. However, they are all focus on the level of syntax, and ignored semantic information. In next section, a new similar query metric will be presented.

3. New Query Suggestion Method Based on Theme and Context

This section presents a new query suggestion method based on Theme and Context. A framework of the study is shown in Figure 1. Firstly, we clustered similar queries. The queries in the same cluster express the same theme. Secondly, we analyze the historical context of queries in session. Finally, the above two factors are integrated for query suggestion. There are two key issues. One is how to measure the similarity of queries, which will be discussed in Section 3.1. The other is how to choose similar query for query suggestion, which will be discussed in Section 3.2.

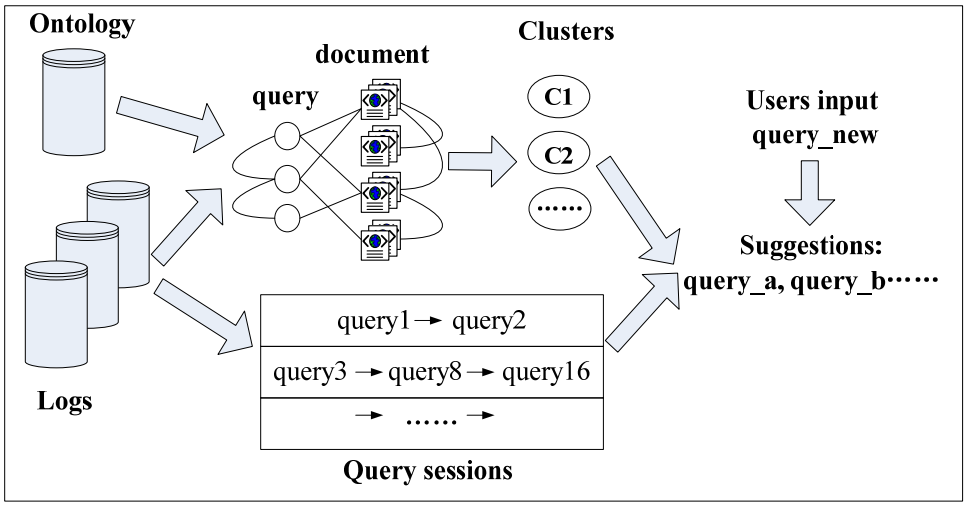

Figure 1. A Framework of the Study

Before discussion, definitions of related concept are as follows:

(1)Term: a basic logical unit that user input in the search engine.

(2)Query: the keywords or expressions that user input in the search engine, including one or more terms. According to the literature on the search engine user behavior analysis it can be found that few users will use logical operator, such as '+', '-', 'and', 'or' or advanced search function to retrieve documents [12-13]. Chinese users to use this feature users only 0.73% [14]. Besides this, it is commonly argued that language semantics are mostly captured by nouns or noun phrases so that the study only focus on noun. So we only extract noun terms for calculation and directly define query as:

$$Query = \{Term_1, Term_2, \ldots\ldots, Term_n\}$$

(3)$U_i(.)$: the set of documents the system presents to the user as search results for the queries $Query_{i.}$

$$U_i(.)=\{d_{i1}, d_{i2}, \ldots\ldots, d_{im}\}$$

3.1. Similar Queries Metric

3.1.1. Similar queries metric based on terms: Similar queries metric based on terms assumes that:

(1)If two queries contain the same terms, they convey the same or similar information needs. The more terms in common, the more similar they are.

(2) If two queries are semantic associated, and the degree of similarity is greater than a certain threshold, they convey the same or similar information needs.

Based on the terms, the new metric is defined as follows:

$$\begin{cases} sim_{term}(Query_p, Query_q) \\ = \frac{N}{\max(N_1, N_2))} + (1 - \frac{N}{\max(N_1, N_2)}) * \frac{sim_{sem}(Queryp', Queryq')}{N_1 * N_2 - N^2} & if \quad Queryp <> Queryq \\ sim(Query_p, Query_q) = 1 & if \quad Queryp = Queryq \end{cases} \tag{1}$$

Where N is the number of the same terms *of* $Query_p$, $Query_q$; N_1, N_2 is the number of terms in $Query_p$, $Query_q$ respectively. $sim_{sem}(Query_p', Query_q')$ is the semantic similarity of the two queries, which denotes in the collection of $Query_p$ (or $Query_q$), after removal the same terms, the semantic similarity of the remains and the terms of $Query_q$ (or $Query_p$). It is defined as:

$$sim_{sem}(Query_p', Query_q') = \sum_{k=1}^{N_1 * N_2 - N^2} x_k \tag{2}$$

Where x_k is the semantic similarity value in the collection *sims* and it is satisfied with $x_k >= x_{k+1}$.

$$Sims = \{x_k \mid x_k = sim_{sem}(Term_{pi}, Term_{qj}), k = 1,2,....., N_1 * N_2\} \tag{3}$$

$$Rank = \{x_k \mid x_k > x_{k+1}\} \tag{4}$$

3.1.2. Similar queries metric based on user clicked documents: Similar queries metric based on user clicked documents assumes that:

(1) If two clicked documents contain the same feature terms, they convey the same or similar information needs and two documents are similar. The more terms in common, the more similar they are.

(2) If two clicked documents don't contain the same feature terms, however the feature terms of the two documents are semantic associated, and the similarity value is greater than a certain threshold, the two documents are similar.

Firstly, according to TF-IDF, feature terms of user clicked documents are extracted. It is commonly argued that language semantics are mostly captured by nouns or noun phrases so that the study only focus on noun.

Then the feature terms of user clicked documents with different queries are compared.

Based on user clicked documents, the similarity of two documents is defined as follows:

$$\begin{cases} sim(d_i, d_j) \\ = \frac{N'}{\max(N_1', N_2')} + (1 - \frac{N'}{\max(N_1', N_2')}) * \frac{sim_{sem}(d_i', d_j')}{N_1' * N_2' - N * N''} & if \quad d_i <> d_j \\ sim(d_i, d_j) = 1 & if \quad d_i = d_j \end{cases} \tag{5}$$

Where N' is the number of the same terms in document d_i, d_j; N_1', N_2' is the number of feature terms in document d_i, d_j respectively, $sim_{sem}(d_i', d_j')$ is the semantic similarity of d_i', d_j', which denotes in the feature terms collection of d_i or (d_j), after removal the same terms, the semantic similarity of the remains and the terms of d_j or (d_i).

$$sim_{sem}(d_i', d_j') = \sum_{k=1}^{N_1'*N_2'-N'^2} y_k \tag{6}$$

Where y_k is the semantic similarity value in the collection $Sims'$ and it is satisfied with $y_k >= y_{k+1}$.

$$Sims' = \{y_k \mid y_k = sim_{semantic}(Term_{pi}', Term_{qj}'), k = 1,2,....., N_1'*N_2'\} \tag{7}$$

$$Rank = \{y_k \mid y_k > y_{k+1}\} \tag{8}$$

Let $U_p(.)$ and $U_q(.)$ be the set of documents the system presents to the user as search results for the queries $Query_p$ and $Query_q$ respectively. The document set that users clicked on for the queries $Query_p$ and $Query_q$ may be seen as follows:

$U_p(.)=\{d_{p1}, d_{p2}, \ldots\ldots, d_{pm}\}$

$U_q(.)=\{d_{q1}, d_{q2}, \ldots\ldots, d_{qn}\}$

Then we need to compute the document similarity matrix:

$$A(\text{Query}_p, \text{Query}_q) = \begin{array}{c} \\ d_{p1} \\ d_{p2} \\ \ldots\ldots \\ d_{pm} \end{array} \begin{array}{c} \begin{array}{cccc} d_{q1} & d_{q2} & \ldots\ldots & d_{qn}. \end{array} \\ \begin{bmatrix} sim(d_{p1}, d_{q1}) & sim(d_{p1}, d_{q2}) & \ldots\ldots & sim(d_{p1}, d_{qn}) \\ sim(d_{p2}, d_{q1}) & sim(d_{p2}, d_{q2}) & \ldots\ldots & sim(d_{p2}, d_{qn}) \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ sim(d_{pm}, d_{q1}) & sim(d_{pm}, d_{q2}) & \ldots\ldots & sim(d_{pm}, d_{qn}) \end{bmatrix} \end{array} \tag{9}$$

Then the similarity of $Query_p$ and $Query_q$ based on user clicked documents is defined as:

$$\begin{aligned} & sim_{doc}(Query_p, Query_q) \\ &= \frac{1}{2} * [\frac{\sum_{i=1}^{m} \max(sim(d_{p_i}, d_{q1}), sim(d_{pi}, d_{q2}), \ldots\ldots, sim(d_{pi}, d_{qn}))}{m} \\ &+ \frac{\sum_{j=1}^{n} \max(sim(d_{p1}, d_{qj}), sim(d_{p2}, d_{qj}), \ldots\ldots sim(d_{pm}, d_{qj}))}{n}] \\ &= \frac{1}{2} * [\frac{\sum_{i=1}^{m} \max_{j=1}^{n}(sim(d_{p_i}, d_{qn}))}{m} + \frac{\sum_{j=1}^{n} \max_{i=1}^{m}(sim(d_{pi}, d_{qj}))}{n}] \end{aligned} \tag{10}$$

3.1.3. Similar Queries Metric based on Multiple Measures: Similarities based on query terms and user clicked documents represent two different points of view. Because user's information needs may be partially captured by each of the above criteria, we would like to define a combined measure that takes advantage of both metric. A simple way to do this is to combine both measures linearly, as follows:

$$\begin{aligned} &sim(Query_p, Query_q) \\ &= k * sim_{term}(Query_p, Query_q) + (1-k) * sim_{doc}(Query_p, Query_q) \end{aligned} \tag{11}$$

Where k is a parameter, which can be adapted manually. Here 0<=k<=1.

If k=0, similar queries metric is based on query terms.

If k=1, similar queries metric is based on user clicked documents.

If 0<k<1, similar queries metric is based on multiple measures.

It is noticed that how to obtain the semantic similarity of two terms is another problem in formula (3) and formula (7). In the paper, we get semantic similarity with the algorithm proposed by Meng [15].

3.1.4. Queries Clustering: Our study obtains the themes by query clustering. There are many clustering algorithms available to us. Because query logs usually are very large, the system does not know how many topics there will be exist. Therefore it is required the clustering algorithm does not need users to set the resulting form of clusters manually, such as the number or the maximal size of clusters.

After comprehensive comparison, the bottom-up hierarchical clustering method is adopted. Firstly, each query is regard as a cluster and the similarity between two clusters is calculated according to formula (12). For any two clusters $Cluser_p$ and $Cluster_q$, cluster function is defined as follows:

$$sim(Cluster_p, Cluster_q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} sim(Query_i, Query_j)}{m \times n} \tag{12}$$

Where $Query_i$, $Query_j$ are any two queries; m is the number of queries in $Cluster_p$; n is the number of queries in $Cluter_q$; $Sim(Query_i,Query_j)$ is the similarity of $query_i$ and $query_j$. If $sim(Clusterp, Clusterq) < \eta$ (η is a threshold value), the clustering process is over.

3.2. Query Suggestion

As mentioned above, similar queries denote the same or similar theme. However, there are many queries in the same cluster. However the search engine page is limited and we only can suggest a certain number of queries for users. For example, baidu and google displays 10 similar queries on the interface. So how to choose query for suggestion is another problem.

In the cluster result, we notice that:

(1) Different clusters may contain the same query. In particular, when there is only one term in the query, it is more prominent. Figure 1 is an example.

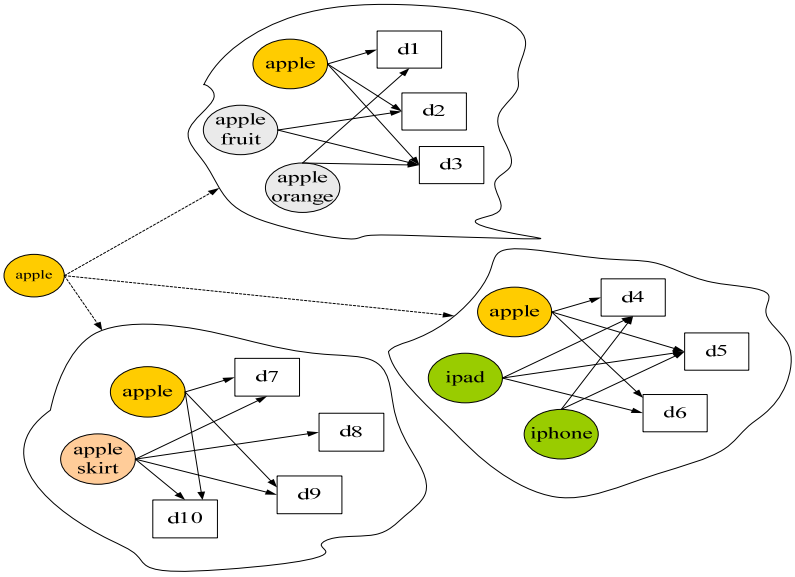

Figure 2. An Example of the Same Query in Different Clusters

This is because different users may execute the same query, and terms in the query may have different meanings. So that the same query are clustered into different clusters, and each cluster represents a similar query theme. For example, different users enter 'apple', but some 'apple' and 'ipad' are clustered to together; some 'apple' and 'apple, orange' are clustered to together.

(2) When a user submits a query, then there may be a continuous queries will be submitted for correcting the initial query. These query sequence constitute query context information for each other, which will contribute to capture the user's query intent.

In this section we will construct query suggestion model based on the two above observations.

3.2.1. Determine query themes: if one query is a member of a cluster, then the meaning of the cluster is the theme of the query, otherwise the similarity is calculated as following:

$$sim\ (query_{new}, Cluster_{k}) = \frac{\sum_{i=1}^{n} sim\ (query_{new}, query_{ki})}{n} \tag{13}$$

Where $query_{new}$ is a new query; $query_{ki}$ is a query in $Cluster_k$; n is the number of queries in $Cluster_k$, and $sim(query_{new}, query_{ki})$ is the similarity of any two queries, which is obtained according to formula(11).

If $sim(query_{new}, query_{ki}) > \eta$, then the meaning of the $Cluster_k$ is the theme of the $query_{new}$. $Cluster_k$ is the theme cluster of $query_{new}$. And the queries in $Cluster_k$ is the candidate queries for suggestion.

3.2.2. Get historical query context information: For any query $query_i$ (1<=i<m), $query_j$(i<j<=m), $query_k$(i<k<=m, j<>k) in the same session: query(s) = $query_1$, $query_2$, …,

$query_m$, these queries in the same session may be similar. If $query_i$, $query_j$, $query_k$ in the same Cluster, then $query_i$, $query_j$, $query_k$ are more similar. Next, let's statistic the frequency fq_j, fq_k that $query_j$, $query_k$ appears in the same session with $query_i$ respectively.

3.2.3. Ranking: Take the queries in the theme cluster that $query_{new}$ belongs to as candidate queries for suggestion and rank them. Related definitions are as follows:

Definition 1: for any query $query_{new}$, the set of queries that appears with $query_{new}$ in the same cluster is C_{new_m} and $|C_{new\_m}|=m$.

Definition 2: in a cluster for any query $query_{new}$, the set of other queries that appears with $query_{new}$ in the same session is C_{new_n} and $|C_{new\_n}|=n$.

Definition 3: in the set of C_{new_m}, after removing the queries contained in C_{new_m}, the set of remaining queries is C_{new_left}, $| C_{new\_left} | = m-n$.

Next, let's ranking the candidate queries.

(1) Rank the queries in C_{new_n}

Set the frequency that $query_j$, $query_k$ appears in the same session with $query_{new}$ as fq_j, fq_k respectively. Rank $query_j$ and $query_k$ order by the values of fq_j, fq_k descending. If $fq_x= fq_y$, rank $query_j$ and $query_k$ order by the values of $sim(query_{new}, queryj_k)$, $sim(query_{new}, query_k)$ descending.

(2) Rank the queries in C_{new_left}

Before ranking, remove some queries.

For any $query_i$ in C_{new_left}, if there is only one term in $query_i$, and the term is the hypernym concept of one term of $query_{new}$ in semantic, remove $query_i$.

For any $query_i$ in C_{new_left}, if $query_i \subset query_{new}$, remove $query_i$.

Next rank the queries in C_{new_left} in the following order.

(1) The queries whose noun is the same with the noun of $query_{new}$ completely. For example 'use, computer' and 'the use of computer'.

(2) The queries that contain $query_{new}$ in word form. For example 'rheumatism, arthritis, difference' and 'rheumatism, arthritis, difference'.

(3) The queries that are hyponyms concepts in semantic of $query_{new}$ completely. For example, 'nuclear weapon' and 'actom bomb'.

(4) The left queries

If there are many such queries in each case, rank the queries order by the similarity values of $query_{new}$ and candidate queries.

3.2.4. Query suggestion: In this section, the suggested queries are generated according to above discussion.

Step1. Take C_{new_n} as $C_{new\_n\_temp}$.

Step2. In each $C_{new\_n\_temp}$, take the top query in turn according to the ranking order and put them into a set $C_{new\_n\_top}$. Rank the queries in $C_{new\_n\_top}$ by the frequency that each query appears in the same session with $query_{new}$. Take the queries in $C_{new\_n\_top}$ for suggestion by the ranking order. If the number of queries suggested is smaller than the total number needed to be suggested, go to step3.

Step3. In each C_{new_n}, remove the queries in $C_{new\_n\_top}$ and take the left queries as $C_{new\_n\_temp}$ and then repeated Step2.

Step4. If the number of queries suggested is equal to the total number needed to be suggested or there is no query in C_{new_n}, the suggestion is over. Otherwise, go to step5.

Step5.Take C_{new_left} as $C_{new\_left\_temp}$.

Step6. In each $C_{new\_left\_temp}$, take the top query in turn according to the ranking order and put them into a set $C_{new\_left\_top}$. Rank the queries in $C_{new\_left\_top}$ by the similarity of each query and $query_{new}$. Take the queries in $C_{new\_left\_top}$ for suggestion by the ranking order. If the number of queries suggested is smaller than the total number needed to be suggested, go to step7.

Step7. In each C_{new_left}, remove the queries in $C_{new\_left\_top}$ and take the left queries as $C_{new\_left\_temp}$ and then repeated Step6.

Step8. If the number of queries suggested is equal to the total number needed to be suggested or there is no query in C_{new_left}, the suggestion is over.

4. Evaluation

In this section, the new method is evaluated by experiments.

4.1. Data Set

For evaluating the performance of our new algorithm, a dataset is necessary. Because of commercial factors, most of search engine would not like to share their query logs. Only the logs of three companies that are Excite, AlltheWeb, AltaVista are available. The latest version is AltaVista_2003. Unfortunately most pages in 2003 are not existed. Therefore, we build a search engine with Nutch, and ask 20 uses in different major to use the search engine randomly. And collect query logs from December 6, 2012 to February 6, 2013. After preprocessing the data, removing incomplete ones, uncivilized ones, a total of 33356 queries and 145152 URLs are left.

4.2. Clustered Results Analysis

The proportions of clustered web queries are shown in Figure 2.

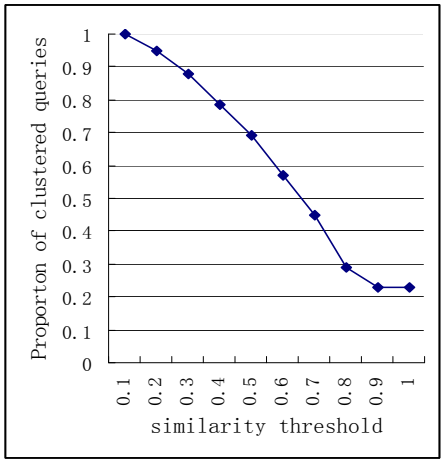

Figure 2(1) k=0

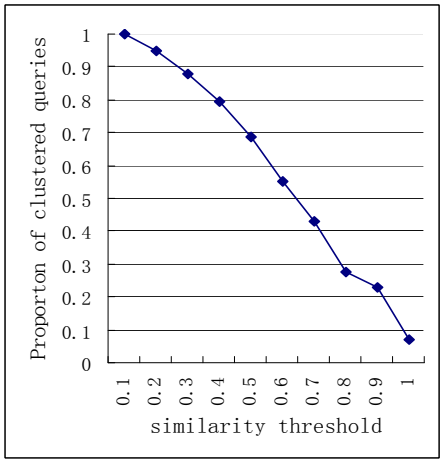

Figure 2(2) k=0.1

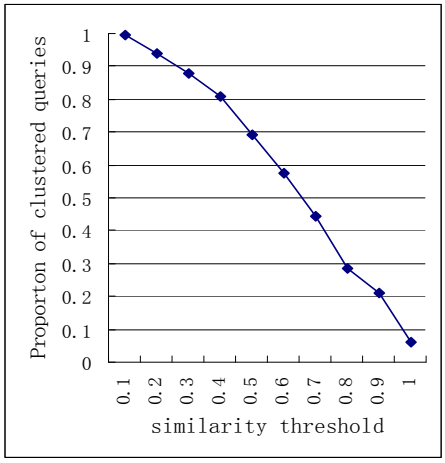

Figure 2(3) k=0.2

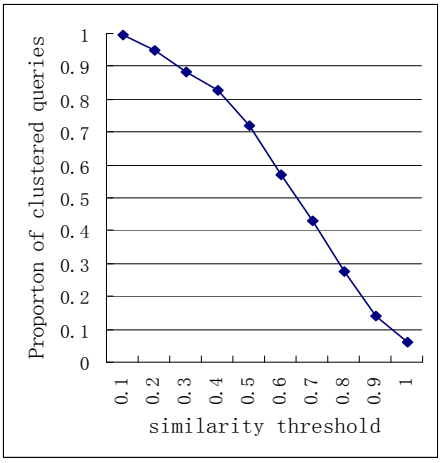

Figure 2(4) k=0.3

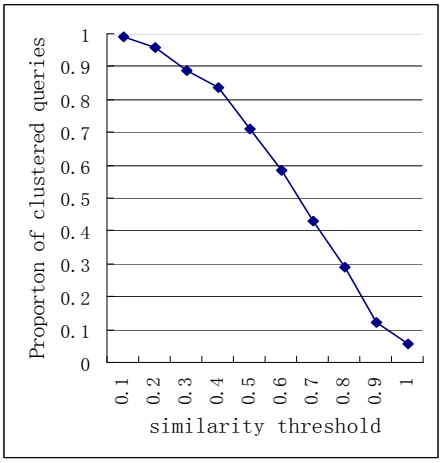

Figure 2(5) k=0.4

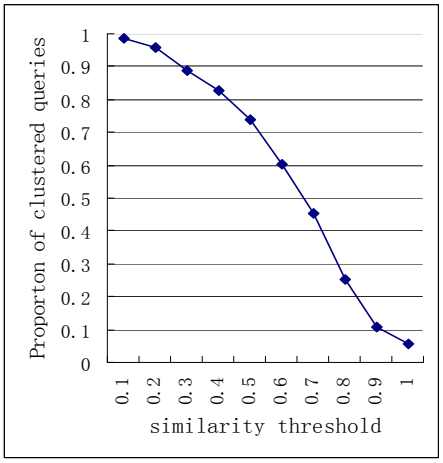

Figure 2(6) k=0.5

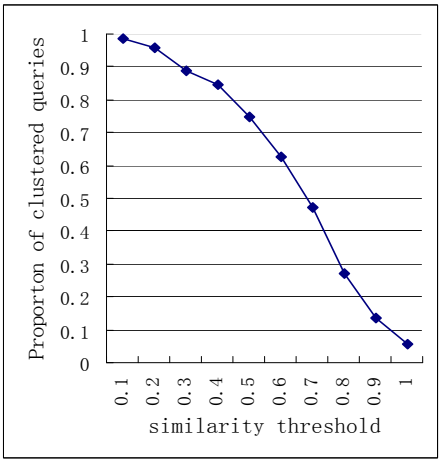

Figure 2(7) k=0.6

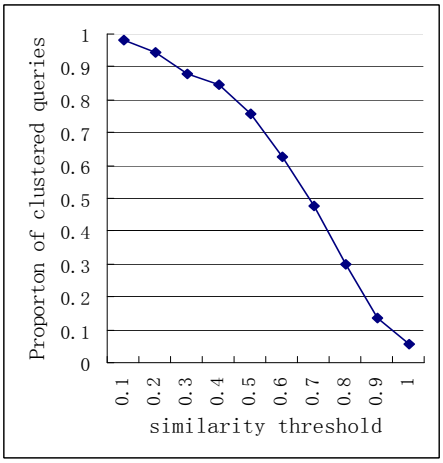

Figure 2(8) k=0.7

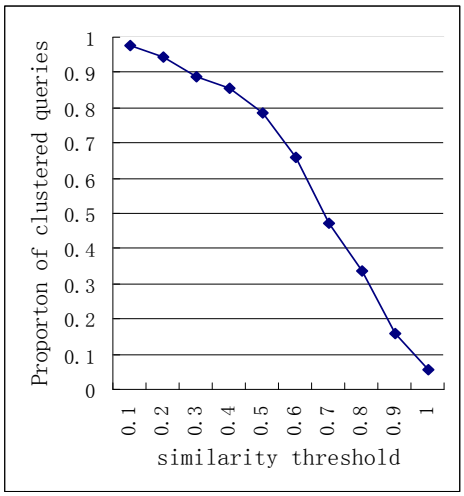

Figure 2(9) k=0.8

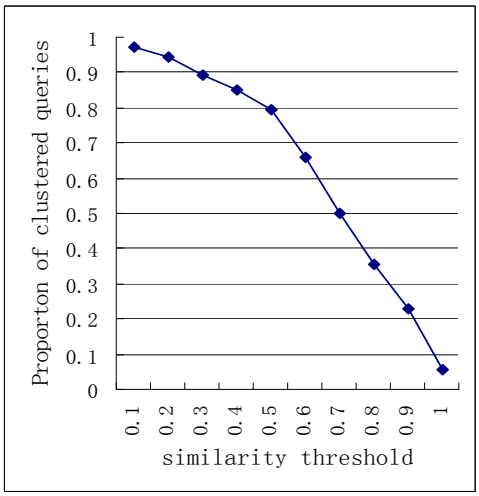

Figure 2(10) k=0.9

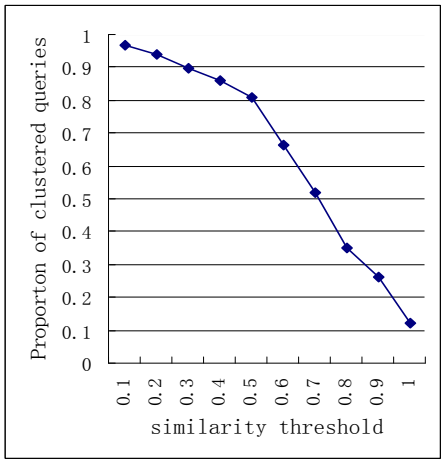

Figure 2(11) k=1.0

From Figure 2, it is noticed that the proportion of clustered queries decreases with the increasing of threshold. After the comparison of clustered result, we find that when k is 0.6 and η is 0.6, the clustered result is more reasonable.

4.3. Query Suggestion Evaluation

In this section, we choose five methods for query suggestion for comparison. These methods are similarity based method, popular query based method, adjacency based method, N-gram based method and the new method proposed in the paper.

In the dataset, 500 queries are chosen randomly and each one is taken as $query_{new}$ respectively. Then 20 undergraduate subjects are asked to judge whether the suggested queries with different methods are similar with $query_{new}$. If it is similar, the suggested query is marked 1, otherwise marked 0. Furthermore, P@N（Precision @ N）is calculated. The result is shown in Table 1.

Table 1. The Precision with Different Methods

methods		P@3	P@5	P@10
Corrected suggestion	New method	98.8%	96.5%	93.4%
	Similarity value based	92.6%	90.9%	87.7%
	Popular query based	90.7%	88.6%	87.9%
	Adjacency based	83.5%	82.4%	81.2%
	N-Gram based	86.7%	84.1%	82.9%
Wrong suggestion	New Algorithm	1.2%	3.5%	6.6%
	Similarity value based	7.4%	9.1%	12.3%
	Popular query based	9.3%	11.4%	12.1%
	Adjacency based	16.5%	17.6%	18.8%
	N-Gram based	13.3%	15.9%	17.1%

For the convenience of comparison intuitively, the compared results of our proposed method with other four methods are provided in Figure 3.

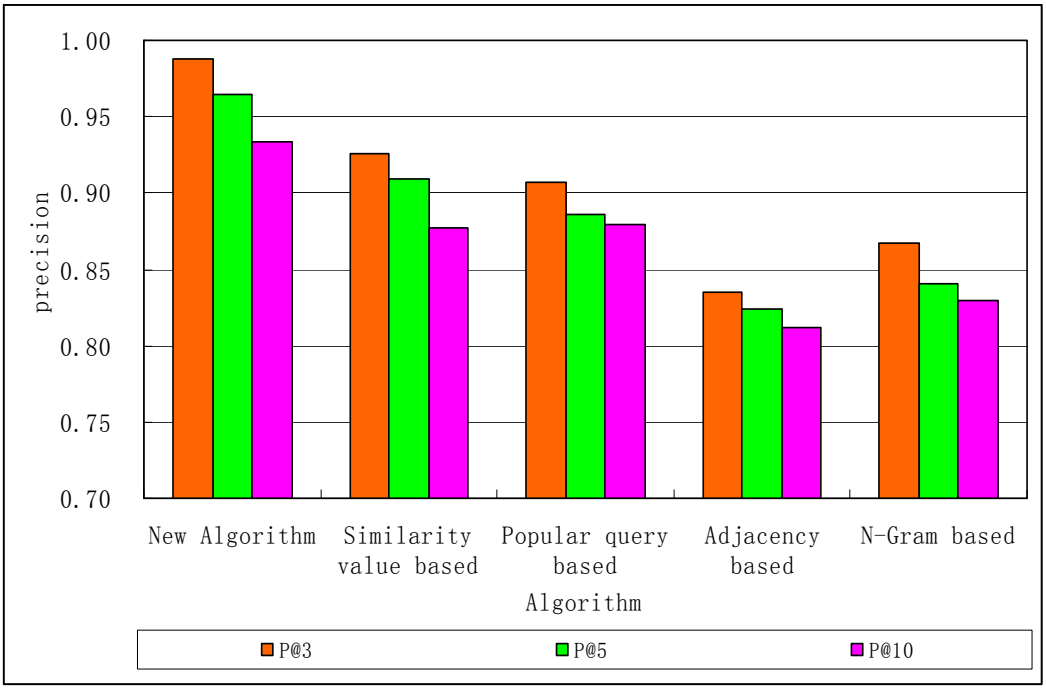

Figure 3. The Comparison of Different Methods

From Table 1 and Figure 3, it can be seen that the precision of suggested query with our proposed method is superior to ones with similarity based method, popular query based method, adjacency based method, N-gram based method. And the highest precision value is get in P@3. All these indicate the good performance of our new method.

5. Conclusion and Future Work

This paper proposes a new query suggestion method which is based on theme and context. Different from previous works, the new method judge similar queries from the level of semantic level and a new framework for similar queries metric is presented. Furthermore, how to choose similar queries for suggestion is discussed and a new query suggestion method is proposed. In the new method, both the themes and the query context are taken into considerate. Finally, we compare the new method with other four methods based on precision by experiments. Experiments show that the new method significantly outperformed than related works. However the study is only a first step in this direction. In future work, we will make some attempt to use the method in intelligent question-answer system of e-learning to find out the FAQs, detect the difficulties of students in e-learning and so on.

References

[1] L. Yanan, W. Bin and L. Jintao, "A Survey of Query Suggestion in Search Engine", JOURNAL OF CHINESE INFORMAT ION PROCESSING, vol. 24, no. 6, **(2010)**.
[2] https: //adwords.google.cn/,**(2013)**, pp. 03-26.
[3] http: //e.baidu.com/pro/, **(2013)**, pp. 03-26.
[4] A. Lee and M. Chau, "The Impact of Query Suggestion in E-commerce Websites", The 10th Annual Workshop on E-Business, **(2011)** December 4, Shanghai, China.
[5] M. A. Hasan, N. Parikh, G. Singh and N. Sundaresan, "Query Suggestion for E-commerce Sites", In Proceedings of the 4th ACM International Conference on Web Search and Data Mining, **(2011)** February 9-12, Hong Kong, China.
[6] J. Jeon, W. B. Croft and J. H. Lee, "Finding Similar Questions in Llarge Question and Answer Archives", In Proceedings of the 14th ACM International Conference on Information and Knowledge Management, **(2005)** October 31-November 5, Bremen, Germany.
[7] P. A. Chirita, C. S. Firan and W. Nejdl, "Personalized Query Expansion for the Web", In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, **(2007)** July 23-27, Amsterdam, the Netherlands.
[8] D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log", Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, **(2000)** August 20 - 23, Boston, MA, USA.
[9] B. M. Fonseca, P. B. Golgher, E. S. de Moura and N. Ziviani, "Using association rules to discovery search engines related queries", Proceedings of the 1st Conference on Latin American Web Congress, **(2003)** November 10-12, Santiago.
[10] J.-R. Wen, J.-Y. Nie and H.-J. Zhang, "Query Clustering Using Content Words and User Feedback", Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, **(2001)** September 9-13, New Orleans, Louisiana, USA.
[11] J. Wang, B. Peng and T. Meng, "Discovering Related Web Gueries Based on Search Engine's User Log", Journal of Beijing University of Posts and TeIecommunications, vol. 28,no. S2, pp. 44-48.
[12] A. Spink, D. Wolfram, M. Bernard and J. Jansen, "Tefko Saracevic. Searching the Web: the Public and Their Queries", Journal of the American Society for Information Science and Technology, vol. 52, no. 3, **(2001)**.
[13] C. Silverstein, M. Henzinger and H. Marais, *et a*,. "Analysis of a Very Large Web Search Engine Query log", In SIGIR Forum, vol. 33, no. 1, **(1999)**.
[14] Y. Huijia, L. Yiqun, Z. Min and R. Liyun, "MA Shaoping. Research in Search Engine User Behavior Based on Log Analysis", JOURNAL OF CHINESE INFORMAT ION PROCESSING, vol. 21, no. 1, **(2007)**.
[15] L. Meng, R. Huang and J. Gu, "An Effective Algorithm for Semantic Similarity Metric of Word Pairs", International Journal of Multimedia and Ubiquitous Engineering, vol. 8, no. 2, **(2013)**.

Authors

Lingling Meng, is an associate professor of Department of Educational Information Technology in East China Normal University. Her research interests include intelligent information retrieval, ontology construction and knowledge engineering.

Runqing Huang, has a PhD from Shanghai Jiao Tong University. He works in Shanghai Municipal People's Government, P. R. China. His present research interests include modeling strategic decisions, economic analysis, electronic government and Logistics.

Junzhong Gu, is Supervisor of PhD Candidates, full professor of East China Normal University, head of Institute of Computer Applications and director of Lab, Director of Multimedia Information Technology (MMIT). His research interests include information retrieval, knowledge engineering, context aware computing, and data mining.