Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Query Recommendation and Its Usefulness Evaluation on Mobile Search Engine

Shunkai Fu[2,1], Bingfeng Pi[1], Michel Desmarais[2], Ying Zhou[1], Weilei Wang[1] and Song Han[1]
[1] Roboo Inc, Suzhou, Jiangsu, P.R.China
[2] Computer Engineering, Ecole Polytechnique de Montreal, Montreal, Quebec, Canada

***Abstract*—In this paper, we study the role of query recommendation on mobile search engine. We start with the discussion of the role of query recommendation in modern search engines. Secondly, a mobile search engine, Roboo® (http://wap.roboo.com), is introduced and we discuss the need for query recommendation over mobile search engine. Thirdly, the query recommendation solution working on Roboo® is introduced in detail, including the models, how they are constructed and how they operate online. Finally, we demonstrate the benefits of query recommendation brought on Roboo® based on the analysis of real search log data collected since its release online in August, 2008. Considering the scarcity of scientific publications about the practical values of query recommendation on commercial mobile search engine, this paper should represent an interesting and useful reference for both academic and industrial colleagues.**

***Keywords*—Mobile search, query recommendation**

I. INTRODUCTION

Mobile search is an emerging promising application in China, especially considering the huge potential user based and the existing based of around 600-million of mobile phone consumers [8]. One latest report from CNNIC (Chinese Internet Network Information Center) indicates that by the end of 2008, the population of mobile phone users that access Internet is about 117 millions [11]. Being a start-up company, Roboo® focuses on providing easy-to-use mobile search service in China, and gains more and more popularity today among mobile "surfers" [1, 2, 3]. Since its introduction online in the late 2006, Roboo® mobile search has been operating for more than 2 years by now, which allows us to accumulate a large amount of real user data that allows us to study the user interest, adjust our product strategy, and evaluate how well we are performing. In this paper, we refer to the traditional search over computer as *wired search*, and the search over a hand-held device as *mobile search*. However, we make no distinction between mobile phones and hand-held devices, notwithstanding that the former is the most widely used equipments for Chinese wireless "surfers". Similarly, wireless search and mobile search are used interchangeably.

Besides relatedness as measured by vector space model [10], various advanced ranking algorithms are being applied that include more factors to re-order the results and improve usage experience, for example, PageRank [5] as we all know. Even so, the level of satisfaction with the pages returned from search engines is still not so high. An independent survey of 40,000 web users found that after a failed search, 76% of them will try to rephrase their queries on the same search engine instead of turning to a different one [7]. This offers an opportunity to effectively improve the search method by recommending related queries formulated by other users of the same search engine. Therefore, query recommendation can potentially employ the "wisdom of crowds" to not only enhance the hit rate of search engines, but also help users to find the target information more quickly and conveniently.

The utilization of query recommendation has been investigated to help users formulate satisfactory queries [7, 12, 13, 14]. However, to the best of our knowledge, there is no published article about the query recommendation on mobile search engine, especially its practical values and real effect. This work extends one previous work by Fu et al. [3] about the cross-channel recommendation over mobile search, but it is meant to be more comprehensive, including both within- and cross-channel recommendations. We also extend the data analysis methods to support our claims.

The contributions of this paper include five aspects. (1) We discuss the features of mobile search, as well as the necessity of providing query recommendation on mobile search. (2) We propose to find related queries, as recorded in query log, based on measures of semantic and non-semantic similarity, and point out their corresponding merits. (3) To make this paper self-contained, we also briefly go through how to recommend collateral information about one queried object, i.e. so-called cross-channel recommendation [3]. (4) We demonstrate how to rank and include the recommendations into mobile search online to provide more convenience and fun to users. (5) We analyze direct effectiveness based on search logs accumulated in the past months.

The rest of the paper is structured as follows. In Section 2, we briefly review Roboo mobile search and list some reasons why query recommendation is desired on our platform. In Section 3, our solution is explained in detail, including the basis, offline processing and online integration/ recommendation. In Section 4, we perform an experimental validation of our techniques using click logs. In Section 5, we give some concluding remarks.

II. QUERY RECOMMENDATION ON MOBILE SEARCH

A. Roboo Mobile Search Engine

Figure 1 (Left) is the home page of Roboo mobile search (http://wap.roboo.com), and it can be accessed via mobile phones anywhere in China covered by wires transmission signal.

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

Figure 1. The home page of Roboo search engine (accessible via http://wap.roboo.com) [Left]. The home page of Roboo News (with hot keywords highlighted with circle) [Right]

Let us briefly explain the Chinese interface. Under the input box, there are 12 different searchable resource repositories, with 3 on each line. They are Game, Theme, Software, Image, MP3, Short Message, Page, Ring, Video, Novel, Mobile Phone and News respectively. They are the primary services provided by Roboo currently. Under them is a list of hot queries, followed by another block, local or life search, like Weather, Bus, Train, Restaurant etc. It can be seen that besides traditional page search, several other vertical search services (categories) are provided on our platform, of which entertainment and resources specific to mobile phones occupies the primary part. These services, and more to come, are designed and deployed based on our understanding about Chinese mobile surfers, and our search log data confirm this strategy. For example, the flow contributed by page search represents less than 20% of total page visit (PV) flow on Roboo.

B. Why We Need Query Recommendation on Mobile Search

Query recommendation has been proved a key factor for the success of modern search engines, and it is deployed by almost all known commercial search engines, like Google, Yahoo! etc [4, 6, 7]. Following are several unique features of mobile search that play as the motivation to emphasize and deploy query recommendation on mobile search engine.

Firstly, typing, or data entry, is quite difficult on mobile phones today. ISO standard keyboard is configured on most mobile phones, such as Nokia's high-end smart phones N95, with only 12 buttons available. Although this compact layout saves considerable space and was acceptable when data entry was mostly limited to numbers, it poses a great challenge today since it requires multi-tap for typing in characters. It is reported that even for skilled users, they can only reach 21-words (in English) per minute; while on PC, the rate is at least 60 words for common users [9]. This situation may prevent users from surfing via mobile phones since navigation requires typing.

Secondly, mobile surfers are often browsing purposelessly when they connect to Internet using mobile phones. In contrast to office use, where we turn to wired search engine, like Google, with a clear purpose, e.g. looking for a paper about query recommendation, we open the open the browser on our mobile phone just to pass the time when we are waiting for a bus, a train, a plane or someone. Our query log also indicates that few queries are about serious events, and most are about entertainment. Given the large span of browsing opportunities just one click away, placing a lonely search box may be not appropriate, especially for mobile surfers because very likely they will turn to others even though we are able to provide tones of interesting images, informative news reports, etc.

Thirdly, again, typing is not easy, which results with shorter queries on mobile search engine. On Roboo, the average length of queries is about 3.5 Chinese tokens (or terms), which places great challenge to a mobile search engine to provide precise result since relevance is highly desired by mobile search users, especially considering small screen (normally less than 3.0 inches) and slow transmission speed factors [2,3] .

Finally, accessing Internet via mobile phone is brand new for most people in China. Although the population using mobile phones to access Internet is about 117 million [11], most of them are still newcomers even though they may be experienced surfers over PC. The shortage of standard mobile phone platform and browsers available on mobile phones, and hard-to-use keyboard make the situation worse. Under this situation, moving the cursor to a highlighted anchor text and clicking on it is the easiest operation because (1) direction and "OK" keys normally are placed at obvious location, and they are most familiar to mobile phone users; (2) response is given right now (by loading a new page) to make the users to feel "safe" and "confident" to go ahead on your site.

Given the above considerations, we focus efforts on means of helping users find entertaining content during boring moments and with the least possible input typing required. To achieve this goal, we spend much effort on the design. For example, on our home page (see Figure 1 Left), a list of hot queries are presented, and piles of information will be retrieved upon clicking, instead of requiring mandatory typing. Figure 1(Right) is another example about the home page of Roboo News, and its design follows the same rule by putting several hot keywords just under the search box. More discussion about the deployment can be found in Section 3.4, and empirical evaluation is presented in Section 4.

III. PROPOSED SOLUTIONS

A. Overall Description

After the coverage of necessary background, including query recommendation, mobile search, and one specific commercial mobile search engine, Roboo, it is time to introduce the query recommendation solution from Roboo.

On Roboo mobile search, three types of recommendation are provided online and offer convenience for mobile searchers:

- Hot queries. They reflect the primary interest of current concern, and they are selected from query log. Hot queries are highly used, and they lead to interesting pages though a single click;
- Related queries given a query. Upon responding a query submitted, the server will return a list of related queries as we see on most search engine today. For each query, it is expanded with a feature vector as extracted from the pages retrieved given the query; then, simple clustering is employed here to group

related queries. This vector is called semantic feature vector, and it has been proposed and successfully applied in several published work [1, 6, 7, 14, 15];

- Suggestion of collateral information, given a query in a specific channel, referred to as cross-channel recommendation. Since there are several channels available on Roboo [1] (text pages, pictures, applications and games, etc), and a given queried object can span many channels, it is useful to do this kind of recommendation. For example, given a query of "Nokia", photos of "Nokia" as well as other resources available for Nokia phones, like game, applications, theme and ring may be attractive to users. It is similar to the notation of cross-sales on Amazon.com or eBay.com.

In the remaining part of Section 3, we will discuss how to find related queries and how to do cross-channel recommendation.

B. Suggest Related Queries within the Same Channel

We summarize the main process of query recommendation in the following steps: Firstly, choosing information sources where most relevant terms as recommendation candidates can be found, given a current query; Secondly, designing a measure to rank candidate terms in terms of relatedness; Finally, utilizing the top ranked relevant terms to reformulate the current query.

Like the main-stream query recommendation working online today, we construct the list of recommended queries given a query based on users' query logs. The similarity between queries is determined based on the proximity of their corresponding so-called semantic feature vector, instead of the query itself. Simple clustering is applied to grouping related queries given the semantic similarity measure.

1) Vector space model

Vector space model (VSM) was first proposed by Salon et al. in 1975 [10]. It is an algebraic model for representing any objects (like documents) as vectors of identifiers (terms in documents). It is the basis for information retrieval, indexing and relevancy computing. In the classical VSM, individual term weights in the document vectors are products of local and global parameters, and the model is known as term frequency-inverse document frequency model (TF-IDF). The weight feature vector for document d of $m-$dimension (that is d consists of m different terms) is $FV(d)=\{w_{d,1},w_{d,2},...,w_{d,m}\}^T$, where $w_{d,t}=tf_t \bullet \log\frac{N}{|\{t\in d\}|}$ and tf_t is term frequency of term t in document d (a local parameter), $\log\frac{N}{|\{t\in d\}|}$ is inverse document frequency (a global parameter). N is the total number of documents in the repository; $|\{t\in d\}|$ is the number of documents containing the term t. Given the normalized feature vector of two objects, O_1 and O_2, their similarity is determined by the cosine of the angle between the vectors: $Sim(O_1,O_2)=\cos\theta=FV(O_1)\bullet FV(O_2)$

2) Semantic Feature Vector and Similarity Measure

Given a query q, and we get its feature vector, denoted as $FV(q)$, with the following sequential steps:

- Issue the query q to a search engine; for now, let us ignore the actual algorithms of the search engine and assume that the approach is generalizable to any engine that provides "reasonable" results;
- Let $D(q)$ be the set of retrieved documents given the query q. In practice, we only keep the top K documents, assuming that they contain enough information, so $D(q)=\{d_1,d_2,...,d_K\}$;
- Compute the TF-IDF term vector tv_i for each document $d_i \in D(q)$, with each element as $tv_i(j)=tf_{i,j}\times\log(\frac{N}{df_j})$ where $tf_{i,j}$ is the frequency of the jth term in d_i, N is the total number of documents that contain the jth term. The TF-IDF is widely applied in the information retrieval (IR) community and has been proved work well in many applications, including the current approach;
- Summing up tv_i, $i=1..K$, to get a single vector. Here, for the same term, its IDF, i.e. $\log(\frac{N}{df})$, is known as identical in different tv_i, so the sum of their corresponding weights can be calculated;
- Normalize and rank the features of the vector with respect to their weight (TF-IDF), and truncate the vector to its M highest weighted terms. What left is the target $FV(q)$.

After we get the feature vector for each query of interest, the measure of semantic similarity between two queries is defined as: $Sim(q_i,q_j)=FV(q_i)\bullet FV(q_j)$. With extracted semantic feature vector, it is possible to find related queries even if no single term is shared between them. For example, given "黄蓉" and "郭靖" (two leading roles in a TV play), the similarity score, e.g. cosine value, between them is 0.0; however, the score becomes 0.96 if we consider their corresponding semantic feature vector, which exactly reflects their true relatedness.

3) Grouping Related Queries via Simple Incremental Clustering

Vector space model and semantic similarity measure play as the basis for measuring how close of two objects, i.e. queries here. The primary procedure can be divided into two phases (see Figure 2):

- Data preparation. A pre-defined number of queries are read from query log first. For each query, check if there is an existing semantic feature vector. If not, submit this query to search engine, and construct its semantic feature vector as described in 3.2.2;
- Clustering and updating. Given semantic feature vectors of a group of queries, do clustering to get a series of clusters. Then, scan the existing cluster database to check if there is any existing cluster that can be merged with any of the new ones. Finally, update the cluster database by adding newly generated clusters or overwriting those merged clusters.

These two phases can be applied to the query log periodically to add newly appearing queries into cluster database. Note that, in each cluster, the similarity score, $Sim(q_1, q_2)$ between each pair of queries is stored, and they are used to rank related queries, i.e. determining which one should appear first (see Figure 3).

In our application, a very greedy but simple clustering algorithm is applied to meet the online usage, and it works as below: (1) Read a query q and its semantic feature vector; (2) If this is the first query to process in this session, generate one cluster, C_1, with only one query q; (3) If not, calculate the similarity score between q and all clusters in memory, $\{C_i\}$ ($i = 1..n$), denoted as $\{Sim_i\}$ ($i = 1..n$); (4) Select the largest score and its index, for example k. If Sim_k is larger than a pre-defined threshold θ, then add q into C_k, and update C_k's centroid; otherwise, generate a new cluster C_{n+1} with C_k's feature vector as its centroid; (5) Repeat step 1-4 for each query to be processed in this session. The merging phases use a similar approach. Same or different threshold value can be defined to determine if merging can be done or not.

In one of our experiment with 50,997 queries and a threshold value as 0.3 (in both clustering and merging steps), we got a total of 13,086 clusters. Among them, 2,168 clusters contain more than 1 query, and the maximum cluster size is 736. On average, each cluster contains 3.90 queries.

C. Cross-channel Recommendation

How to realize the cross-channel recommendation in a quick and effective manner online will be addressed in this section. The overall procedure is divided into the following steps:

- Index queries as they appear in each channel's search log separately, by filtering out those queries with no results. New queries can be added into their corresponding channel's index online;
- Given $< q, ch >$, i.e. query q submitted in a specific channel ch, do a search over each channel's query index as prepared in step 1;
- If there is no exact match, go to step 4; else, $q_{rec} = q$. Then, if q appears in more than one channel, select the channel with highest frequency of this query; otherwise, directly go to step 6;
- Filter out recommended candidates with frequency lower than pre-defined threshold value θ_1 and those with similarity score lower than θ_2;
- Randomly select a recommended query as q_{rec} if there is at least one match by the end of step 4. Otherwise, $q_{rec} = null$;
- If $q_{rec} \neq null$, retrieve q_{rec}'s corresponding channel label, ch_{rec}, to construct the final recommendation $< q_{rec}, ch_{rec} >$. On the user interface, this recommendation appears as "If you are looking for q_{rec}'s ch_{rec}", e.g. "If you are looking for Apple's image."
- When "If you are looking for q_{rec}'s ch_{rec}" is clicked, a search about q_{rec} is done in channel ch_{rec}, and the retrieved results are presented to the searcher.

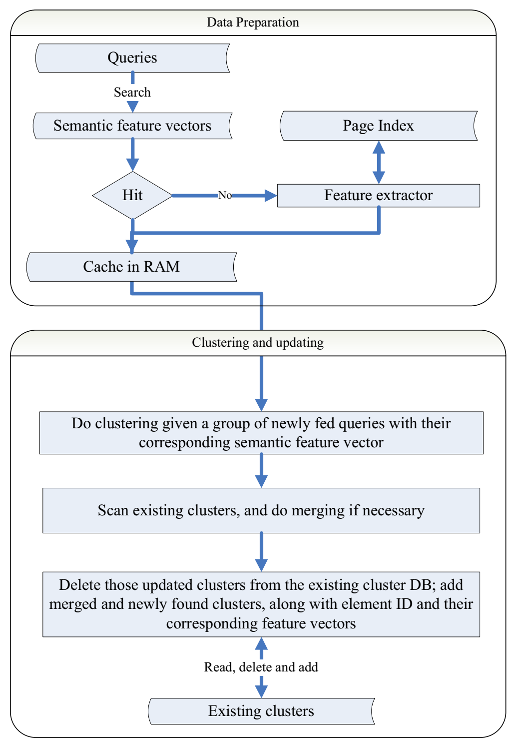

Figure 2. The overall procedure to do query clustering

From this brief description, it can be seen that two kinds of information are necessary for the cross-channel query recommendation: (1) Search log data. It records queries and their corresponding frequencies; (2) The similarity between the active query (i.e. the one being studied) with any other queries existing in the search log. This can also be calculated with

$Sim(q_1, q_2)$, i.e. the proximity between the two semantic feature vectors corresponding to the two queries of interest.

D. Online Deployment and Recommendation

1) Hot Queries

Figure 1 demonstrates our strategy of presenting hot queries hyperlinks. Normally, hot queries are slightly modified by editors to make them more concise or descriptive. Inappropriate queries, like porn queries, have to be filtered first.

2) Related Queries within a Channel

While working online, a series of queries will be retrieved upon seeing a submitted query. On Roboo, the similarity score, $Sim(q_1, q_2)$ is used to rank the recommendations. Figure 3 is one example of query recommendation. Given "手机搜索"(mobile search)，a series of famous mobile search engines available in China are recommended, including Roboo, Baidu, iAsk etc.

3) Cross-channel Recommendation

Cross-channel recommendation is deployed online as well, and one example is shown in Cross-channel recommendation is deployed online as well, and one example is shown in Figure 4. Given a query "诺基亚"(Nokia) in Page channel, a cross-channel recommendation is given by our algorithm as shown in Figure 4 (highlighted with circle in Left): 想找"诺基亚"主题 (If you are searching for Nokia's Theme). If this suggestion is clicked, the user will be led into our Theme channel with a series of theme resources suitable to install on Nokia's phones (Figure 4, Right) (Note that Theme is something like skin, allowing customizing the appearance).

4) Expected Ideal Usage Scenario: Search without Any Typing

The need for query recommendation is discussed in Section 2.2. Here, we will illustrate one expected ideal usage scenario to happen on Roboo mobile search: *Michel is waiting for a bus to home. He takes out his Nokia N95 smart phone, and clicks one previously saved bookmark to visit http://wap.roboo.com (Roboo mobile search). On the home page, he notices Olympics Opening Ceremony (hot query) on the home page, which incites him to click it. A list of news report about Beijing Olympics Opening Ceremony is presented. Below that, he sees "Phelps"(query recommendation) and clicks without hesitation since he wants to know how many gold medals Phelps has won. A list of news about Phelps is retrieved and presented on Michel's phone. This time, other than those news reports, he is attracted by "Are you looking for Phelps' photos"(cross-channel recommendation). Upon clicking that, photos about Phelps engaging in Olympics match are returned.*

Although this is a fictitious scenario, it is what we expect of the effect of the query recommendation deployed, i.e. users of Roboo mobile search can find something entertaining solely by clicking.

IV. Performance Analysis based on Query Log

A. Hot Queries

In our log, we will record page-visit (PV) from our recommendation page where not only hot queries, but other edited resources are presented for easy clicking and browsing. However, we don't further distinguish the PV contributed by hot queries and others, therefore, there is no way to exactly tell how much the hot queries contribute. Nevertheless, it is still possible to infer something from Figure 5 in which the percentage of "Recommendation" PV among the total PV is listed, from May 2008 to Feb 2009. It can be seen that over 25% of our total PVs come from clicking of recommended resources as shown in Figure 1.

B. Related Queries within a Channel

Clicking of those recommended queries is recorded in our query log. In this study, we use the data ranging from August 4, 2008 to January 11, 2009, totally 23 weeks. Figure 6 shows the ratio of queries recommended and clicked over the total number of queries processed in each week: (1) The ratio ranges from 8.8% to 20.5%, with 13.6% as the average level; (2) The overall trend is that the ratio is increasing with time on.

C. Cross-channel Recommendation

Clicking of those cross-channel recommendations is marked with special tag as well in our query log for study purpose. Figure 7 shows the ratio of cross-channel recommendation clicked to the total number of queries processed in each week, from August 4, 2008 to January 11, 2009. On average, this ratio is about 3.2%. So, together with query recommendation, cross-channel recommendation contributes totally about 16.8% among the total queries received, about one every six query comes from query recommendation. If we add hot queries into consideration, the proportion will be even higher. Actually, if we count the extra contribution of PVs by users after they are led to a new channel upon clicking cross-channel recommendation, then a higher percentage of PVs is expected from the contribution of query recommendation as we discuss. However, due to tracing limitations, this study is not done.

V. Conclusion

This study provides new light on query recommendation on mobile search. Firstly, we give a brief background information about mobile search in China and the query recommendation on wired search. Then, we discuss why query recommendation is more demanded on mobile search compared with wired search. Following that, the query recommendation solution implemented on Roboo mobile search is discussed, including hot query, query recommendation within a query and cross-channel recommendation. We also demonstrate how they are deployed on Roboo mobile search and an expected ideal usage scenario. We believe a successful query recommendation solution depends not only the algorithm of finding what to recommend, but also on the user experience: the manner in which to integrate recommendations into user interface. Finally, we study the effect, i.e. usefulness, of the query recommendation based on months of query logs.

In view of the scarcity of work on the topic of query recommendation on mobile search, we hope this early investigation will provide a valuable reference for both academic and industrial colleagues.

REFERENCES

[1] Fu, S., Desmarais, M.C.., Pi, B., Zhou, Y., Wang, W., Zou, G., Han, S. and Rao, X. 2008. "Simple but effective porn query recognition by k-NN with Semantic Similarity Measure". In Proceedings of APWeb/WAIM (Suzhou, P.R.China, April 1-4, 2009).

[2] Fu, S., Pi, B., Han, S., Guo, J., Zou, G. and Wang, W. 2008. "User-centered solution to detect near-duplicate pages on mobile search engine". In Proceedings of SIGIR Workshop on Mobile Information Retrieval (MobIR) (Singapore, 2008).

[3] Fu, S., Pi, B., Zhou, Y., Desmarais, M.C., Wang, W., Han S. and Rao, X. 2008. "Cross-channel query recommendation on commercial mobile search engine: Why, how and empirical evaluation". In Proceedings of 13th Pacific-Asia KDD (PAKDD) (Bangkok, Thailand, April 27-30, 2009).

[4] Liu,Z. and Sun, M. 2008. "Asymmetrical query recommendation method based on bipartite network resource allocation". In Proceedings of WWW (Beijing, P.R.China, April 21-25, 2009).

[5] Document,L., Brin,S. and Winograd,T. "The PageRank citation ranking: Bringing order to the web". Stanford Digital Libraries Working Paper. 1998.

[6] Baeza-Yates, R., Hurtado, C. and Mendoza M. 2004. "Query recommendation using query logs in search engines". In Proceedings of International Workshop on Clustering Information over the Web (ClustWeb, in conjunction with EDBT, Greete, Greece, March, 2004).

[7] Beeferman, D. and Berger,A. 2000. "Agglomerative clustering of a search engine query log". In Proceedings of ACM SIGKDD.

[8] http://www.cnetnews.com.cn/2008/0725/1012706.shtml (in Chinese)

[9] Silfverberg, M., MacKenzie, I.S. and Korhonen, P. 2000. "Predicting text entry speed on mobile phones", In Proceedings of SIGCHI Conference on Human Factors in Computing Systems (The Hague, The Netherlands, 2000).

[10] Salton, G., Wong, A. and Yang, C.S. 1975. "A vector space model for automatic indexing". Communications of the ACM, vol 18, no.11, pages 613-620.

[11] http://www.cnnic.com.cn/html/Dir/2009/01/12/5447.htm (in Chinese)

[12] Chirita, P.A., Firan, C.S. and Nejdl, W. 2007. "Personalized query expansion for the web". In Proceedings of SIGIR Conference on Information Retrieval.

[13] Cui,H., Wen,J.-R.,Nie,J.-Y and Ma, W.-Y. "Query expansion by mining user logs". IEEE Transaction of Knowledge Data Engieering, vol 15, no.4, pages 829-839, 2003.

[14] Wen,J.-R., Nie, J.-Y. and H.Zhang. "Query clustering using user logs". ACM Transaction of Information System, vol 20, no.1, pages 59-81, 2002.

[15] Sahami,M. "Mining the web to determine similarity between words, objects, and communities". In Proceedings of the 19th FLAIRS Conference, 2006.

Figure 3 One example of query recommendation. By default, following the research result, four recommendations are listed (highlighted with circle, left). By clicking ">>", more are displayed on a new page (Right).

Figure 4. One example of cross-channel recommendation (Left, Page channel), and user will be led into a Theme channel with a list of theme resource retrieved (Right), if click the recommendation.

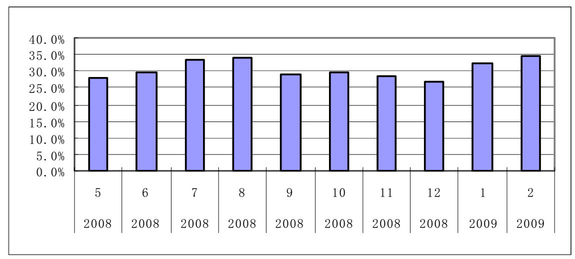

Figure 5. Percentage of "recommendation" PV relative to the total PV, from May 2008 to Feb 2009.

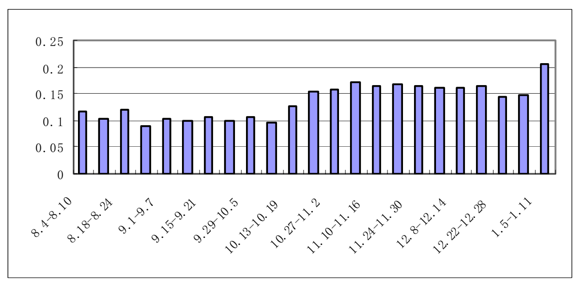

Figure 6. Percentage of clicking of recommended queries relative to the total amount of queries processed, from Aug 4,2008 to Jan 11, 2009, totally 23 weeks.

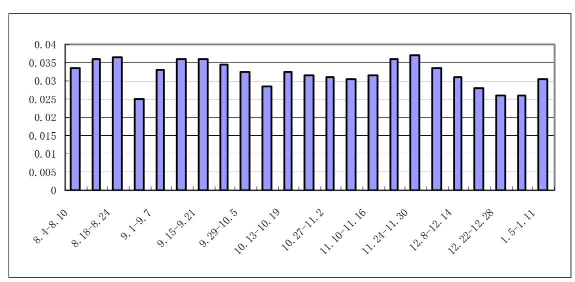

Figure 7. Percentage of clicking of cross-channel recommendation relative to the total amount of queries processed, from Aug 4,2008 to Jan 11, 2009, totally 23 weeks.