An Effective Recommendation System for Querying Web Pages*

Chien-Liang Wu and Jia-Ling Koh†
Department of Information and Computer Education
National Taiwan Normal University
Taipei, Taiwan 106 R.O.C.
E-mail: jlkoh@ice.ntnu.edu.tw

Abstract

In this paper, a recommendation system for querying web pages is developed. When users query web pages through a search engine, the query keywords, browsed web pages, and feedback values are collected as user query transactions. A clustering algorithm based on bipartite graph analysis is designed to determine clusters of query keywords and the browsed web pages, called access preference clusters. Next, association rules of query keywords and web pages are mined for each access preference cluster. The feedback values of browsed web pages are incorporated into the calculation of the support and confidence for each association rule to reflect the subjective opinions. Based on the mined clusters and rules, the system applies the concept of collaborative filtering to recommend highly semantics relevant web pages in the access preference clusters partially covered by a user profile or a given query. The initial experiment result shows the system can improve the querying effect of searching engines.

1. Introduction

Data on Internet is increasing rapidly. The World Wide Web has become one of the important sources to obtain data. To help users find useful data from the World Wide Web in a fast and effective manner, search engines provide users query interfaces to search relevant Web pages links by inputting keywords or query terms. By applying the indexing strategies of Web pages on contained keywords, most search engines return a large number of satisfied page links .However, most of the time, only a small proportion among the returned results is feasible web pages for users, not limited to contain the queried keywords exactly, is an important issue in a search engine.

Data clustering strategies are often applied to perform data analysis, decision-making, data retrieving, image segmentation, etc.[2][4][8]. Recently, many researchers [3][6][7][9] also focused on the clustering methods for analyzing user browsing behaviors to obtain clusters of web pages or the associations among query keywords and web pages.

The correlation of query keywords and web pages was deduce in [9] by performing a clustering method. In this approach, semantic relevance of query keywords was analyzed according to query keywords and accessed web page links which represented the similar information to user feedbacks in a traditional IR environment.

User profiles are crucial information for analyzing user clusters in [11]. Mostly, a user profile consists of personal data, transaction data, transaction behavior, preferred data, and classification data of the user. In [12], not only the browsed pages, the browsing frequency, time, and order of browsing activities of users in a specific site were also taken into consideration to determine user groups of similar browsing behaviors.

A most popular technique applied in recommendation system is collaborative filtering. This type of recommendation systems [5][7][10] performed data clustering method to group users according to their preference or behaviors. The recommendations for a user are determined based on the behavior histories of users in the same group with the user.

Most related works discussed above used user browsing histories to show the preferences of users. In addition, collaborative filtering strategy was used to recommend web pages or documents most popularly. However, in the results of most clustering analysis on user preferences, one user only belongs to a certain cluster. For users having various kinds of preferences, the number of users in the same cluster may be very small, resulting in a limited range of recommendable data. That is, other recommendable information provided by users with partially similar preferences gets lost. Moreover, among the returned results of a query on a search engine, users usually browsed the first few page links. Therefore, only considering whether or not a returned page link is browsed is insufficient to determine the semantic association between a query keyword and the page.

In this paper, a recommendation system for querying web pages is developed. When users query web pages

* This work was partially supported by the Republic of China National Science Council under Contract No. 92-2213-E-003-012
† Author to whom all correspondence should be addressed.

through a search engine, the query keywords, browsed web pages, and feedback values are collected as user query transactions. A clustering algorithm based on bipartite graph analysis is proposed to determine clusters of query keywords and the browsed web pages, called access preference clusters. Next, association rules of query keywords and web pages are mined for each access preference cluster. The feedback values of browsed web pages are incorporated into the calculation of the support and confidence for each association rule. Based on the mined clusters and rules, the system applies the concept of collaborative filtering to recommend highly semantics relevant web pages in the access preference clusters partially covered by a user profile or a given query.

The organization of the rest of this paper is as follows. Definitions of the terms mentioned throughout the paper are introduced in Section 2. Section 3 describes the strategy for mining associations between query keywords and the browsed web pages. A recommendation method for querying web pages through collaborative filtering approach is presented in Section 4. In Section 5, the experiment shows the applicableness of the proposed system. Finally, Section 6 concludes this paper.

2. Basic definitions

In this paper, a recommendation system is designed to be cooperated with search engines to provide more feasible results for web documents searching. The basic idea is to mine the association between user queries and Web pages according to querying and browsing histories of users.

The system proposed in this paper consists of three major processing components. Data preprocessing is performed as described in this section. The other parts will be introduced in detail in the following two sections.

In this proposed system, a user is allowed to log in as a registered member or anonymous user. The system applies JSP sessions to distinguish access behaviors of users. A unique session number is given for each connection, which remains valid for 1.5 hours. Users log in the system to query the required Web pages by inputting keywords. Moreover, the registered members can assign feedback values after browsing the recommended Web pages. Therefore, in addition to the usual access information (such as the IP address, domain of requesting machine, time and date of request, access method, IP address of requested file, etc.), the member ID, session numbers, query keywords, and feedback values are also recorded in the log file of the recommendation server for each document access. First of all, data in log files is preprocessed to extract information relevant to browsing behaviors of individual users.

For each document access, the extracted information includes user ID, section number, query keyword, http of browsed web page, and feedback value. A *user query session* consists of a query keyword and the set of corresponding browsed web pages and feedback values in a user query session.

User query sessions collected by the system within certain period are transformed into user query transactions. A unique transaction id is assigned for each user query transaction. An example of user query session shown in Table 1 is transformed into user query transactions as shown in Table 2. It shows user “User1” inputs query keyword “q1”, then browses the returned pages d1, d2, and d3, and gives relevant feedback values 1.0, 1.0, and 0.6 for these pages respectively. The constructed user query transaction is denoted as <q1> (d1 1.0) (d2 1.0) (d3 0.6). In addition, the system records user query sessions with the same User ID into profile of the corresponding registered user.

[Definition 1] Let UT be a set of user query transactions, Q be the set of query keywords in UT, and D be the set of web pages in UT. E is a relationship between Q and D such that $E \subseteq Q \times D$.

$\forall$ T $\in$ UT, T contains $\{< q_i > (d_j, f_j)\} \rightarrow (q_i, d_j) \in E$.

A *browsing association graph* G(UT)=(Q, D, E) represents a relationship of keyword set Q and web page set D, which is a bipartite graph. Figure 1 shows the browsing association graph for the running example shown in Table 2.

[Definition 2] Given a browsing association graph G(UT)=(Q, D, E). Vertices q_i and d_j are *adjacent* if $q_i \in Q$, $d_j \in D$, and $(q_i, d_j) \in E$. For each $q_i \in Q$, the *neighbors* of q_i are the set of web pages adjacent to q_i, denoted as **Neighbor(q_i)**, where $\text{Neighbor}(q_i) = \{d_k | d_k \in D \wedge (q_i, d_k) \in E\}$. In Figure 1, q_3 is adjacent to vertices d_1, d_2, and d_5. Thus, Neighbor(q_3)={d_1, d_2, d_5}.

[Definition 3] The similarity between keyword q_i and q_j, denoted as **Sim(q_i, q_j)**, is evaluated by the following formula:

$$\text{Sim}(q_i, q_j) = \frac{|\text{Neighbor}(q_i) \cap \text{Neighbor}(q_j)|}{|\text{Neighbor}(q_i) \cup \text{Neighbor}(q_j)|}.$$

The function is also extended for evaluating the similarity between two sets of keywords, in which the neighbor of a set of keywords Q is defined as:

$$\text{Neighbor}(Q) = \bigcup_{q_i \in Q} \text{Neighbor}(q_i).$$

Association rules are usually used to represent associations that occur between data items in a transaction database. In this paper, in order to provide feasible web pages searching, association between a keyword and the set of browsed pages is analyzed. In addition, data items are supposed to have various association weights in a transaction. In a user query transaction, the association weight for the query keyword is set to 1.0. Besides, the

Table 1. Example of user query session

UserID: User1	
aaa-BgcJm-5856	<q1> d1 1.0
	<q1> d2 1.0
	<q1> d3 0.6
	<q2> d2 0.8
	<q3> d1 0.8
	<q3> d5 1.0
	<q4> d4 0.8
	<q4> d5 0.8
	<q4> d6 0.2
	<q5> d5 0.6
	<q5> d6 0.2
UserID: User2	
adj-BhkSy-8118	<q1> d3 0.2
	<q2> d1 0.8
	<q2> d2 1.0
	<q3> d2 1.0
	<q3> d5 1.0
	<q4> d4 0.6
	<q4> d5 0.8
	<q5> d5 0.8
	<q5> d6 0.2
Anonymous	
abe-GnmOst-1758	<q1> d2 0.8
	<q1> d3 0.2
	<q2> d1 0.6
	<q2> d3 0.2
	<q3> d1 0.8
	<q3> d5 1.0
	<q4> d4 0.8
	<q4> d6 0.2
	<q5> d5 1.0

Transfer →

Table 2. Example of user query

TID	User Query Transaction
1	<q1> (d1 1.0) (d2 1.0) (d3 0.6)
2	<q2> (d2 0.8)
3	<q3> (d1 0.8) (d5 1.0)
4	<q4> (d4 0.8) (d5 0.8) (d6 0.2)
5	<q5> (d5 0.6) (d6 0.2)
6	<q1> (d3 0.2)
7	<q2> (d1 0.8) (d2 1.0)
8	<q3> (d2 1.0) (d5 1.0)
9	<q4> (d4 0.6) (d5 0.8)
10	<q5> (d5 0.8) (d6 0.2)
11	<q1> (d2 0.8) (d3 0.2)
12	<q2> (d1 0.6) (d3 0.2)
13	<q3> (d1 0.8) (d5 1.0)
14	<q4> (d4 0.8) (d6 0.2)
15	<q5> (d5 1.0)

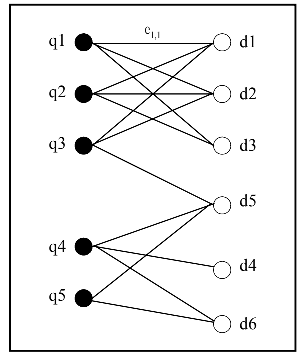

Figure 1. Bipartite graph

feedback value of each browsed Web page represents the association weight of the page in the transaction. For each data item X, X.weight(T_i) denotes the association weight of X in transaction T_i. X.weight(T_i)=0 if X is not contained in transaction T_i. In the example shown in Table 2, d_2.weight(T_2) = 0.8, d_2.weight(T_3) = 0 and q_2.weight(T_2) = 1.0

[Definition 4] Let **sup(X)** denote the support of data item X in the database,
$\text{sup}(X) = \sum_{i=1}^{N} X.\text{weight}(T_i) \Big/ N$, in which N is the number of user query transactions in the database.

[Definition 5] Let **sup(X, Y_1,…,Y_k)** denote the support of data itemsets {X, Y_1,…,Y_k}.
$\text{sup}(X, Y_1,\ldots, Y_k) = \sum_{i=1}^{N}\left(X.\text{weight}(T_i) * \prod_{j=1}^{k} Y_j.\text{weight}(T_j)\right) \Big/ N$, in which N is the number of user query transactions in the database.

[Definition 6] The support and confidence of association rule $X \rightarrow Y_1,\ldots,Y_k$ ($k \geq 1$) are denoted as **sup** $(X \rightarrow Y_1,\ldots Y_k)$ and **conf** $(X \rightarrow Y_1,\ldots Y_k)$ respectively, which are computed by the following two formulas.
$\text{sup}(X \rightarrow Y_1,\ldots Y_k) = \text{sup}(X, Y_1,\ldots, Y_k)$, and
$\text{conf}(X \rightarrow Y_1,\ldots Y_k) = \frac{\text{sup}(X, Y_1,\ldots, Y_k)}{\text{sup}(X)}$.

3. Association mining for query keywords and web pages

3.1. Access preference clusters

3.1.1. Clusters analysis. The similarity function formulated in definition 3 is used to calculate the similarity value between two query keywords. That is, the keyword clusters are evaluated according to the degree of neighborhood overlapping in the browsing association graph. If two query keywords have a similarity value no less than a predefined threshold value, these keywords are allocated in the same cluster.

```
Access Preference Clustering Algorithm
Input: browsing association graph G, similarity-θ,
        and element-θ
Output: Sets of access preference clusters
  Assign each query keyword in G to a cluster;
  Repeat
    Compute the similarity values for each pair of clusters;
    Select the largest Sim(Ci, Cj) among the cluster pairs;
    If Sim(Ci, Cj) is greater than similarity-θ
      Then  Merge clusters Ci and Cj;
    Else Exit Repeat loop
  End Repeat
  Remove the clusters whose number of elements are below
  element-θ.
```

Figure 2. Algorithm of clustering analysis

Figure 2 is the pseudo code of the clustering algorithm for finding keyword clusters. Suppose similarity threshold value (denoted as *similarity-θ*) is set to 0.4, and threshold value of element numbers (denoted as *element-θ*) is set to 2. In the running example, two clusters of keywords C1={q_1, q_2, q_3}and C2={q_4, q_5} are obtained. The keywords in a cluster and the corresponding browsed web pages are jointly called an *access preference cluster* as the example shown in Figure 3. It is possible that a web page is allocated in more than one access preference just like d5 shown in Figure 3.

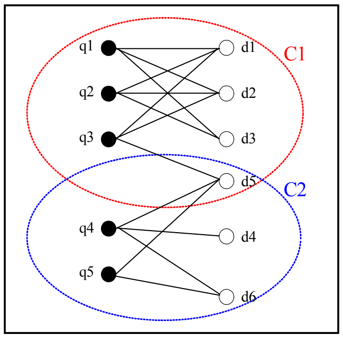

Figure 3. Example of access preference cluster

3.2. Association rules of keywords and web pages

3.2.1. User query transactions partitioning. After access preference clusters are found, user query transactions are correspondingly partitioned according to the clusters of query keywords. For example, in Figure 3, cluster C_1 includes query keywords q_1, q_2, and q_3, and cluster C_2 includes query keywords $q_4$and q_5. Therefore, the user query transactions are partitioned into two corresponding parts, SUT_1 and SUT_2, respectively. SUT_1 contains user query transactions TID {1, 2, 3, 6, 7, 8, 11, 12, 13} and SUT_2 contains {4, 5, 9, 10, 14, 15}.

In our approach, the access preference clusters are analyzed according to the relationship between query keywords and browsed web pages. Thus, it is possible that a user profile contains multiple preference clusters. In the running example shown as Table 1, the query keywords contained in the user profile of "User1" is {q_1, q_2, q_3, q_4, q_5}. It shows that the access preferences of User1 belong to both clusters C_1 and C_2. This approach reflects multiple access preferences of a user in order to better catch user preferences in actual cases.

Our approach aimed to determine association rules of the type $q_i \rightarrow D$, where q_i is a single query keyword and D is a set of web pages. In order to extract the representation information in a access preference cluster, frequent itemsets in the corresponding partition of user query transactions are mined by applying Apriori algorithm [1]. However, the support of a data item is calculated according to definitions 4 and 5. Since only the association between a single query keyword and web pages is considered, there is no need to find frequent itemsets that include more than one query keyword. Suppose the minimum support θ_s is 0.15. By following the example shown in Table 2, the frequent itemsets in partition SUT_1 is as shown in Table 3, in which L_{k-1} denotes the set of frequent itemsets consisting k-1 elements.

For each frequent itemset, an association rule is generated and rule confidence is calculated according to definition 6. Only the association rules with confidence no less than the minimum confidence θ_c are retained. Let the set of association rules deduced from the partition of query transactions of *i*th access preference cluster be denoted as $Rset_i$. Table 4 shows the set of association rules ($Rset_1$) deduced from the frequent itemsets shown in Table 10 (minimum confidence θ_c =0.4).

Table 3. Frequent ietmsets

L_1	Sup	L_2	Sup	L_3	Sup
q1	0.33	q1, d2	0.2	q3, d1, d5	0.17
q2	0.33	q2, d1	0.16		
q3	0.33	q2, d2	0.2		
d1	0.44	q3, d1	0.17		
d2	0.51	q3, d5	0.33		
d5	0.33	d1, d2	0.2		
		d1, d5	0.17		

Table 4. Association rule for SUT_1

Association Rule	Conf.
q1→d2	0.6
q2→d1	0.48
q2→d2	0.6
q3→d1	0.51
q3→d5	1.0
q3→d1, d5	0.51

4. Collaborative recommendation system

Two types of web pages recommendation methods are provided by this proposed system: (1) recommendation based on user profile, and (2) recommendation based on inputted query keywords.

4.1. Personalized recommendations

This type of recommendation method is provided for registered members. When a member submits a recommendation request, the system will find the access preference clusters that have keywords contained in the user's profile. The confidence of an association rule is used as the *recommendation value* of web pages in the rule. Thus, the association rules between keywords and web pages in such an access preference cluster are sorted by their confidences. The web pages in the first n rules, that the user has not yet browsed, are recommended in that order.

Figure 4 shows an example of access preference clusters C_1, C_2, and C_3 mined out, and their association rule sets for query keywords and web pages are $Rset_1$, $Rset_2$, and $Rset_3$, respectively. Suppose the user profile of "user3" is as shown in Table 5. After user3 submits a recommendation request, the user profile shows that user3 belongs to access preference clusters C_2 and C_3. The system then determines the n association rules with most significant confidences from $Rset_2$ and $Rset_3$, individually. Suppose n is set to 3. The set of rules {$q_5 \rightarrow d_3$, $q_4 \rightarrow d_5$, $q_5 \rightarrow d_6$} from $Rset_2$, and {$q_7 \rightarrow d_{11}$, $q_6 \rightarrow d_4$, $q_8 \rightarrow d_{10}$} from $Rset_3$ will be selected. Finally, the recommended web pages are d_3, d_6, d_{11}, d_4, and d_{10}. The web page is ignored from the result because it has been browsed by user3.

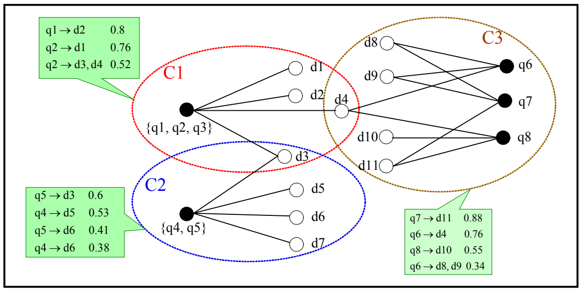

Figure 4. Example of access preference clusters

Table 5. Profile of "User3"

User ID: User3	
aaa-BgcJm-5856	<q4> d5 0.8
	<q4> d7 0.6
	<q5> d7 1.0
aks-HesBq-1475	<q6> d8 0.4
	<q6> d9 0.6
	<q7> d8 0.8
	<q7> d9 0.8

4.2. Recommendations based on query keywords

The second method of recommendation applies to both members and anonymous users. Each user submits a query keyword to the system for querying semantics related web pages. Based on the access preference cluster C_i the given keyword belongs to, the association rules in $Rset_i$ are sorted by confidence. Web pages in the first significant n rules are recommended.

Continue the example shown in Table 5. Suppose a user inputs query keyword q_6 and requires a recommendation search, the three rules with most significant confidences are selects from the association rules set $Rset_3$ because q_6 belongs to access preference cluster C_3. The web pages in these rules, namely d_{11}, d_4, and d_{10}, are then recommended. By applying this method, the semantics relevant web pages which do not contain the given keyword also can be obtained, such as d_{10} in this example.

5. System implementation and experiment

5.1. System implementation

The system is performed on a 800MHz Pentium III PC with 256MB of memory, running Windows 2000 Professional and Resin-2.1.0 Web Server. VB programming language is used to implement the internal training tasks of the system. JSP programming language is used to implement the functions of on-line querying recommendations.

Users log in the system as members or anonymous users. The system provides three functions for users: searching web pages through Chinese Yahoo searching engines, personalized recommendations of web pages, and recommendations of web pages by giving query keyword. The personalized recommendations are provided only for members since the corresponding user profiles are required for this function.

Users can submit query keywords to Chinese Yahoo searching engine and receive searched results through our system. Due to that a large amount of information may be returned, the system displays only the first 200 web page links returned to users. The system also provides a brief description for each returned page link. After clicking the title of a web page, users can further browse the full content of an article. Moreover, the feedback button located on the right ride of the title is used to assign a feedback value for the browsed page. This system uses the Liker's five-point scale to divide the feedback values into five levels (1 to 5). The corresponding feedback values for web pages are 0.2, 0.4, 0.6, 0.8, and 1.0. The default value is set to be 0.6. The system will record querying history of each user in the user profile, including query keywords, browsed web pages, and feedback values. The collected data is transformed to user query transactions. The clustering analysis and association mining are performed on these transactions, as introduced in the previous sections, to provide the other two functions.

5.2. Experiment

Due to privacy concerns, the access logs of searching engines cannot be obtained easily. The experiment is performed on the students in database laboratory to show the effects of query recommendation provided in this system.

In order to sure to find access preference clusters, query keywords used in this experiment were limited to data mining related terms, including "data warehouse", "OLAP", "data mining", "sequential mining", "decision tree", "association rule", "collaborative filtering", and "personalization". A total of 1861 query browsing records were collected, in which 1681 query records were served as training data and the other 180 query records were used as testing data.

The mean absolute error (MAE) is defined as the following formula to evaluate the feasibility of recommended web pages for a user's query.

$$MAE_i = \frac{\sum_{s=1}^{r} |p_s - f_s|}{r}$$

in which p_s and f_s denote the recommendation value of the system and the feedback value given by the user for web page s, respectively. In addition, r denotes the number of web pages browsed by the user. The overall MAE value for a set of query recommendations is the average of the MAE values for these recommendations.

The smaller MAE value implies the recommendations fit the requirements of users.

When training the system, the values of *similarity-θ* and *element-θ* are set to 0.02 and 2, respectively. Two query preference clusters are extracted, in which the included keywords are shown as the follows Table 6.

Table 6. Result of clusters

Cluster 1	sequential mining, collaborative filtering
Cluster 2	data mining, data warehouse, OLAP

In addition, when mining the association rules in SUT_1 and SUT_2, the minimal supports are set to 0.15 and 0.09, and 65 rules and 43 rules mined out from $Rset_1$ and $Rset_2$, respectively. By changing the number of recommended web pages, the MAE values calculated from the testing data are shown in Figure 5. The result shows that the recommendations meet the requirements of users very closely. As the number of recommend web pages increases, the mean absolute error decreases even more.

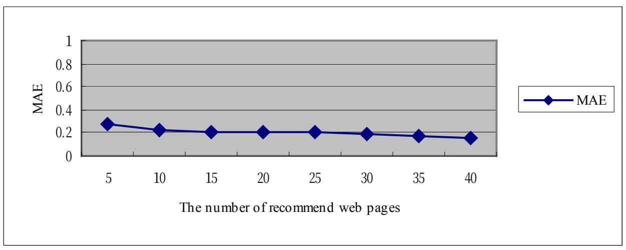

Figure 5. Result of MAE measure

6. Conclusion and future research directions

In this paper, a recommendation system cooperated with Chinese Yahoo searching engine is proposed. A clustering algorithm is designed to determine access preference clusters from previous user behaviors. These clusters represent the semantics related associations between query keywords and browsed web pages. According to the submitted query keywords or profiles of users, feasible results are recommended based on the association rules of keywords and web pages mined from user behaviors in each cluster. In this approach, a user with various preferences usually belongs to multiple access preference clusters. Therefore, the browsing behaviors of users with partially similar preferences are also used to provide recommendation information when applying collaborative filtering strategies. Moreover, the feedback values of web pages given by users are used in the computations of supports and confidences for association rules to reflect the subjective opinions. The initial experiment result shows the system can improve the querying effect of searching engines.

In the future, the plan is to obtain web server logs from search engines or collects user querying behaviors in much longer period in order to get more user query transactions for training system widely. In addition, the feedback values of browsed web pages are considered to be included in the similarity computation of keywords when mining access preference clusters. Furthermore, browsing time and frequency will be considered as weights and considered in the confidence computations of association rules in order to further improve the effectiveness of this system.

References

[1] R. Agarwal, and R. Srikant, "Fast Algorithm for Mining Association Rule in Large Databases," in Proceeding of The 20th International Conference on Very Large DataBases, 1994.

[2] D. Barbara, and P. Chen, "Using the Fractal Dimension to Cluster Datasets", in Proceeding of the 6th International Conference on Knowledge Discovery and Data mining, ACM SIGKDD, 2000.

[3] D. Beeferman, and A. Berger, "Agglomerative Clustering of A Search Engine Query Log," in Proceedings of the 6th International Conference on Knowledge Discovery and Data Mmining, ACM SIGKDD, 2000.

[4] C.H. Cheng , A.W. Fu, and Y. Zhang, "Entropy-based Subspace Clustering for Mining Numerical Data," in Proceeding of International Conference on Knowledge Discovery and Data Mining, ACM SIGKDD, 1999.

[5] X. Fu, J. Budzik, and K.J. Hammond, "Mining Navigation History for Recommendation," in Proceeding of International Conference on Intelligent User Interfaces, ACM, 2000.

[6] H.J. Kim, and S.G. Lee, "A Semi-Supervised Document Clustering Technique for Information Organization," in Proceeding of the 9th International Conference on Information and Knowledge Management, 2000.

[7] B. Mobasher, R. Cooley, and J. Srivastava, "Creating Adaptive Web Sites Through Usage-Based Clustering of URLs," in Proceedings of the Workshop on Knowledge and Data Engineering Exchange, 1999.

[8] K. Wang, C. Xu, and B. Liu, "Clustering Transactions Using Large Items," in Proceeding of the 8th International Conference on Information and Knowledge Management, ACM, 1999.

[9] J. Wen, J.Y. Nie, and H.J. Zhang, "Clustering User Queries of A Search Engine," in Proceedings of the 10th International World Wide Web Conference, 2001.

[10] Y.H. Wu, Y.C. Chen, and A.L.P. Chen, "Enabling Personalized Recommendation on the Web based on User Interests and Behaviors," in Proceeding of International Workshop on Research Issues in Data Engineering , RIDE IEEE, 2001.

[11] K.L. Wu, C.C. Aggarwal, and P.S. Yu, "Personalization with Dynamic Profiler," Advanced Issues of E-Commerce and Web-Based Information Systems, in Proc. of the 3th International Workshop on Computer Networks and Mobile Computing (WECWIS), 2001.

[12] J. Xiao, and Y. Zhang, "Clustering of Web Users Using Session-Based Similarity Measures," in Proceedings of International Conference on Computer Networks and Mobile Computing, 2001.