Expert Systems with Applications 27 (2004) 365–377

Expert Systems with Applications

www.elsevier.com/locate/eswa

Effective personalized recommendation based on time-framed navigation clustering and association mining

Feng-Hsu Wang[a,*], Hsiu-Mei Shao[b]

[a]Department of Computer and Communication Engineering, Ming Chuan University, 5 Teh-Ming Road, Gwei Shan District, Taoyuan County 333, Taiwan, ROC

[b]Department of Information Management, Ming Chuan University, 5 Teh-Ming Road, Gwei Shan District, Taoyuan County 333, Taiwan, ROC

Abstract

Personalized recommendation by predicting user-browsing behavior using association-mining technology has gained much attention in web personalization research area. However, the resulting association patterns did not perform well in prediction of future browsing patterns due to the low matching rate of the resulting rules and users' browsing behavior. This research proposes a new personalized recommendation method integrating user clustering and association-mining techniques. Historical navigation sessions for each user are divided into frames of sessions based on a specific time interval. This research proposes a new clustering method, called HBM (Hierarchical Bisecting Medoids Algorithm) to cluster users based on the time-framed navigation sessions. Those navigation sessions of the same group are analyzed using the association-mining method to establish a recommendation model for similar students in the future. Finally, an application of this recommendation method to an e-learning web site is presented, including plans of recommendation policies and proposal of new efficiency measures. The effectiveness of the recommendation methods, with and without time-framed user clustering, are investigated and compared. The results showed that the recommendation model built with user clustering by time-framed navigation sessions improves the recommendation services effectively.

Keywords: Personalized recommendation; Web usage mining; Clustering; Association mining; Collaborative filtering; Web-based learning environment

1. Introduction

Internet has stirred the fast development of web sites equipped with rich resources in a variety of application sectors. However, users are often apt to get lost in such an environment due to its complicated structure and huge amount of information. Therefore, a new design method that can adapt a web site to user needs is of great importance to improve the usability and user retention of the web site. Such an adaptation feature, called web personalization (Mulvenna, Anand, & Buchner, 2000), will become a fundamental feature of future web systems, and their success heavily relies on the system's capability to anticipate user needs, and respond properly. In particular, personalized recommendation is one form of web personalization that could find important applications in e-business (such as Amazon.com and google.com) and e-learning sectors. In the context of personalized recommendation, resources (web pages, products, advertisements, etc.) are recommended to a user according to the inner-established knowledge model that anticipates the user's needs. In this paper, we focus on the personalized recommendation of web pages that are adapted according to the access patterns constructed by analyzing user navigation information (Fu, Budzik, & Hammond, 2000).

In the WWW context, web sites are generating a great amount of web usage data that contain useful information about users' behavior. The term 'Web Usage Mining' (Cooley, Mobasher, & Srivastava, 1997) was introduced by Cooley et al., in 1997, in which they define web usage mining as the 'automatic discovery of user access patterns from Web Servers'. Web usage mining has gained much attention in the literature as a potential approach to fulfill the requirement of web personalization (Cooley et al., Eirinaki & Vazirgiannis, 2003; Fu et al., 2000; Gery & Haddad, 2003; Mobasher, Cooley, & Srivastava, 2000; Mulvenna et al., 2000). The discovered knowledge indicating users' navigational behavior is useful for the system to personalize the web site according to each user's behavior and profile.

* Corresponding author. Tel.: +886-3-350-7001x3446; fax: +886-3-329-4449.

E-mail addresses: fhwang@mcu.edu.tw (F.-H. Wang), smshao@mcu.edu.tw (H.-M. Shao).

0957-4174/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2004.05.005

The data mining methods that are employed including association rule mining, sequential pattern discovery, clustering and classification. In this paper, we focus on the association-mining method, which is a widely used data analysis method in web usage mining (Gery & Haddad; Lee, Kim, & Rhee, 2001; Mobasher et al.; Wang & Thao, 2003).

Two measurements are often used to evaluate the effectiveness of the prediction knowledge obtained through the data analysis methods. One is the accuracy, and the other is the coverage (Gery & Haddad, 2003). The accuracy measures the system ability to provide correct predictions, while the coverage measures the system ability to provide predictions for the testing database and/or future unseen user cases. A third criterion is the recall rate that measures the system ability to provide as many correct predictions as users need. According to the evaluation results of (Gery & Haddad), the accuracy and coverage rate of the association-mining technique is usually quite low. (Note that their results also showed though the sequence mining method produced higher accuracy than association mining did, it produced much lower coverage.) The association patterns did not perform well in prediction of future navigation patterns due to the low matching rate of the prediction rules and users' navigation behavior. Our previous study (Wang & Thao, 2003), applying the association mining over all users' navigation sessions to establish a knowledge model to predict users' next request in an e-learning web site, also revealed similar evidences to this fact. This drawback shows the limit of the prediction knowledge built only through conventional association-mining technique.

Therefore, this research explores a new personalized recommendation method integrating user clustering and association-mining techniques. Instead of performing the association-mining task over all users' navigation sessions, users are first clustered elaborately so that users in each cluster demonstrate shared navigation characteristics. For this purpose, this paper proposes a new user-clustering scheme based on time-framed navigation sessions. The assumption is that each user may have very diverse preferences at different time stages of his/her visits to a web site. This is especially true for e-learning/training applications. Therefore it might help to better represent and analyze the diverse navigation characteristics when viewing the user's navigation history in a more fine-grained viewpoint. So a concept of 'time-framed' navigation sessions is proposed, that is something like the sampling technique widely used in digital image processing. Historical navigation sessions for each user are divided into frames of sessions based on a specific time interval. Selection of a good time interval is an elaborative decision that depends on the characteristics of the applications. This research proposes a clustering method, called HBM (Hierarchical Bisecting Medoids Algorithm) to cluster users based on the time-framed navigation sessions. Those time-framed navigation sessions in the same group are then analyzed using the association-mining method to establish a recommendation model for similar students in the future.

Finally, the personalized recommendation method is applied as part of an intelligent navigation guider in an e-learning web site, and different plans of recommendation policies and effectiveness measures are investigated and evaluated. The effectiveness of the recommendation methods, with time-framed user clustering under different time intervals, are investigated and compared. The results showed that the recommendation model built with user clustering by time-framed navigation sessions provides effective recommendation services.

The subsequent sections are organized as follows. Section 2 depicts basic facts about association mining, clustering and personalized recommendation systems. In Section 3, we present the main idea and design of the personalized recommendation method based on time-framed navigation clustering and association mining. Section 4 describes the various recommendation policies and gives a description of the performance criteria used in evaluating the recommendation effectiveness. Section 5 depicts the application of the personalized recommendation method to an e-learning web site. In Section 6, experiments on the real-world data sets are conducted to evaluate the variant recommendation policies with the performance criteria. Finally, we make some remarks on the limitations of the method and portray some future work.

2. Backgrounds

2.1. Recommendation systems

In a large-scale distributed network environment like Internet, the popularization of computers and the Internet have resulted in an explosion in the amount of digital information. As a result, it becomes more important and difficult to retrieve proper information adapted to user preferences. Therefore, personalized recommendation systems are in need to provide proper recommendations based on users' requirements and preferences (Mulvenna et al., 2000; Riecken, 2000) In general, there are two types of recommendation systems, the content-based filtering systems and the collaborative filtering systems (Mobasher et al., 2000; Nichols, 1997).

2.1.1. Content-based filtering systems

Content-based filtering techniques are based on content analysis of target items. For examples, the technique of term frequency analysis for text document and its relation to the user's preferences is a well-known content analysis method. In content-based filtering systems, recommendations are provided for a user based solely on a profile built up by analyzing the content of items that the user has rated in the past and/or user's personal information and preferences. The user's profile can be constructed by analyzing the responses to a questionnaire, item ratings, or the user's navigation information to infer the user's preferences and/or interests. However, a pure content-based filtering system has several shortcomings and critical issues remained to be solved, including that only a very shallow analysis of specific kinds of content (text documents, etc.) are available and that users can receive only recommendations similar to their earlier experiences, and the sparseness problem of item rating information (Kwak & Cho, 2001; Lee et al., 2001).

2.1.2. Collaborative filtering systems

In collaborative filtering, items are recommended to a particular user when other similar users also prefer them. The definition of 'similarity' between users depends on applications. For example, it may be defined as users having similar ratings of items or users having similar navigation behavior. This kind of recommendation systems is the first one that uses the artificial intelligence technique to do the personalized job (Riecken, 2000). A collaborative filtering system collects all information about users' activities on the web site and calculates the similarity among the users. If some users have similar behavior, they will be categorized to the same user group. When a user logins into the web site again, the system will first compute the group most similar to the user using methods like the *k*-nearest neighborhood, and then recommend items that the members of the group prefer to the user. A pure collaborative filtering system also has several shortcomings and critical issues, including that the coverage of item ratings could be very sparse, hence yielding poor recommendation efficiency; and that it is difficult to provide services for users who have unusual tastes, and the user clustering and classification problems for users with changing and/or evolving preferences (Konstan et al., 1997). Table 1 shows a brief comparison between the two filtering methods.

2.2. Data mining

Data mining, which is also referred to as knowledge discovery in database, is a process of nontrivial extraction of implicit, previously unknown and potentially useful information (such as knowledge rules, constraints, regularities) from data in database (Chen, Han, & Yu, 1996) The data mining algorithms can be divided into three major categories based on the nature of their information extraction: predictive modeling (also called classification or supervised learning), clustering (also called segmentation or unsupervised learning), and frequent pattern extraction (Agrawal, Imielinski, & Swami, 1993). In the following, we briefly review some of the mining methods that are relevant to our research.

2.2.1. Association mining

Association mining is one of the most well-studied mining methods in data mining (Agrawal et al., 1993; Agrawal & Srikant, 1994; Chen et al., 1996; Han & Kamber, 2001). It serves as a useful tool for discovering correlations among items in a large database. It explores the probability that when certain items are present, which other items also present in the same affairs. An association rule is a condition of the form $X \Rightarrow Y$ where X and Y are two sets of items. An interpretation of the association rule in a business trade situation is when a customer buys items in X, the customer will also buy items in Y.

There are two important threshold values used in mining association rules: support and confidence. Support indicates the frequencies of the occurring patterns in the rule. In the minimum support approach, association rules are generated by discovering large itemsets. A set of items X is called a large itemset if the support rate of X, with respect to a transaction database meets the minimum support requirement. Confidence denotes the strength of the implication of the association rule. If the confidence is higher, the rule is more reliable.

2.2.2. Clustering

Clustering is a useful technique for discovering interesting data distributions and patterns in the underlying data. It is a process of grouping physical or abstract objects into

Table 1
Comparison between content-based filtering and collaborative filtering systems

	Content-based filtering	Collaborative filtering
Advantage	A user can receive proper recommendations without helps from other users	A user may have chances to receive items that s/he never contacts before, but may be of his/her potential interests
	It is more feasible to tackle the problems of multiple user interests and interest transference by monitoring the change and evolving of user profiles	Facilitate the sharing of knowledge and/or experiences among users having similar interests
Limitation	Some types of items (e.g. multimedia) are not easy to analyze	It is hard to provide recommendations for users that have unusual preferences
	A user can just receive items that are similar to his/her past experiences	It is hard to cluster and classify users with changing and/or evolving preferences

classes of similar objects. Clustering analysis helps construct meaningful partitioning of a large set of objects based on a 'divide and conquer' methodology which decomposes a large-scale system into smaller components to simplify design and implementation (Chen et al., 1996). The principle of clustering is maximizing the similarity inside an object group and minimizing the similarity between the object groups.

The most well-known and commonly used partitioning methods are *k*-means, *k*-medoids, and their variations (Han & Kamber, 2001). In the *k*-means algorithm, cluster similarity is measured in regard to the mean value of the objects in a cluster, which can be viewed as the cluster's center of gravity. The *k*-means method, however, can be applied only when the mean of a cluster is defined. This may not be the case in some applications, such as when data with categorical attributes are involved. Besides, it is sensitive to outliers since an object with an extremely large value may substantially distort the distribution of data. On the other hand, instead of taking the mean value of the objects in a cluster as a reference point, the *k*-medoids method use the medoid, which is the most centrally located object in a cluster. Therefore, the *k*-medoids method takes advantage over the *k*-means in the aspects of versatileness and outlier insensitivity. However, the necessity of both methods for users to specify *k*, the number of clusters, in advance can be seen as a common disadvantage.

2.3. Recommendation systems based on association rules mining technologies

As data mining techniques become more and more maturing, researchers have explored their applications in recommendation systems in the last decade, trying to improve the efficiency and the effectiveness of the recommendation systems. Among those efforts, Fu et al. (2000) try to integrate the collaborative filtering method and association-mining technology to develop a recommendation system called SurfLen that recommends web pages on the web site. Their research reorganized the web pages collected from the 'Yahoo!' search engine, and experimented on the influence of the noise upon the recommendation effectiveness (Fu et al.). Besides, Lee et al. integrate the collaborative filtering method and association-mining technology to develop a recommendation system to recommend movies for the audiences on the MovieLends web site (http://www.movielens.umn.edu) (Lee et al., 2001).

3. The personalized recommendation method

Fig. 1 shows the framework of our personalized recommendation scheme. The recommendation module builds a knowledge base of navigation patterns by first clustering users based on the time-framed navigation sessions over the historical navigation database, and then establishes the access patterns for each user group using the association-mining technique. To produce personalized recommendations for a user, the group most similar to the user's navigation sessions is first selected, and then the recommender applies the prediction rules in the corresponding rule base to generate the item recommendation list that sorts the items in terms of relevance.

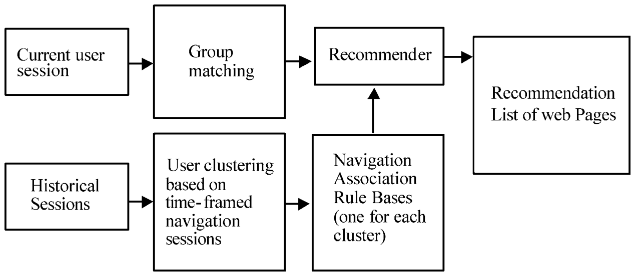

Fig. 1. The personalized recommendation mechanism.

3.1. User clustering based on time-framed navigation sessions

Instead of performing the association-mining task over all users' navigation sessions, which might eliminate the visibility of important access patterns, users are clustered elaborately by sampling the navigation sessions in a specific time frame. Historical navigation sessions for each user are divided into frames of sessions based on a specific time interval. Selection of a good time interval is an elaborative decision that depends on the characteristics of the applications. For example, in this study context, the web-based virtual classroom environment, candidate time intervals may be a 'week' or a 'semester', which coincides with the teaching/learning schedule in Taiwan. Fig. 2 shows two possible framings of navigation sessions based on the week and semester frame, respectively.

A long time interval, such as a 'semester', provides a macro view of a user's navigation behavior embedded with richer access information, but it may be hard to generalize the navigation rules in such a macro behavioral view. On the other hand, a shorter time interval provides a micro view on a user's navigation behavior, but it may lose some important access information. Hence, the impacts of time frame intervals on the recommendation effectiveness will be a main issue investigated in this paper. Users are clustered based on these time-framed navigation sessions. Those framed navigation sessions in the same group are then analyzed using the association-mining method to establish an association rule base as the recommendation model for

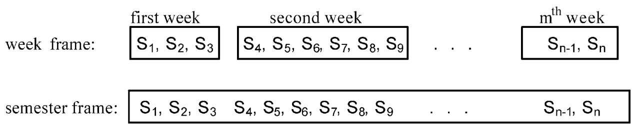

Fig. 2. Two framing schemes of a user's navigation sessions.

similar students in the future. In the following we present a clustering method, called HBM (Hierarchical Bisecting Medoids Algorithm) to cluster users based on the time-framed navigation sessions. One feature of this algorithm is that it avoids the common problem of requiring users to pre-specify on the number of clusters by using a hierarchical clustering technique.

3.1.1. The HBM clustering algorithm

The algorithm combines features of the k-medoids and hierarchical clustering. We will provide a new definition of user similarity based on the concept of time-framed navigation sessions in Section 3.1.2. The algorithm is outlined as follows:

Step 1.
Set the minimal intra-cluster similarity: δ (a user-specified parameter).
Step 2.
Initially, there is only one cluster, consisting of all objects (each object is represented as a set of time-framed user navigation sessions).
Step 3.
For each cluster of objects, compute its medoid as described below:
Step 3-1.
Initially, select a medoid randomly from the objects within the cluster.
Step 3-2.
Calculate the average similarity between the tentative medoid and the other objects within the cluster.
Step 3-3.
Apply the algorithm of swapping medoids in the k-medoids algorithm, and find the new medoid that results in the maximal average similarity.
Step 3-4.
Repeat Steps 3-2 to 3-3, until no new medoid can be found.
Step 4.
For each cluster i, calculate the average intra-cluster similarity s_i, where $s_i, = Avg_p\{Sim(G_i, p)\}$, G_i is the medoid of cluster i, p is an object in cluster i, and $Sim(\,)$ is a similarity function.
Step 5.
If $s_i < \delta$, apply the 2-medoid algorithm to divide cluster i into two sub-clusters, and repeat Steps 2 to 4; Otherwise stop.

3.1.2. User similarity with time-framed navigation sessions

As mentioned above, users' navigation sessions are divided into frames of navigation sessions according to a pre-specified time interval. Given two users U_i and U_j, and one of their time-framed navigation sessions, as shown below, respectively,

$U_i : TF_u(U_i) = \{S_{i1}, S_{i2}, \ldots, S_{in}\}$, the uth time-framed sessions,

$U_j : TF_v(U_j) = \{S_{j1}, S_{j2}, \ldots, S_{jm}\}$, the vth time-framed sessions,

where session S_k is a collection of web pages that the users have visited during a session. Actually, the framed navigation sessions represent the user's navigation behavior during a specific time interval. When we say that users U_i and U_j are similar to each other at two time intervals, it means the two users have similar navigation behavior during the periods of time intervals (but not necessarily the same time interval, the time intervals may not be overlapped). Doing this with a good choice of time interval, we hope to be able to discover more useful navigation patterns that can be used to improve the recommendation effectiveness. Specifically, first define the similarity of two session records, S_{is} and S_{jt}, as follows:

$$Sim(S_{is}, S_{jt}) = \frac{|S_{is} \cap S_{jt}|}{|S_{is} \cup S_{jt}|}, \qquad 1 \leqq s \leqq n,\ 1 \leqq t \leqq m. \quad (1)$$

Next, define the similarity of the two time-framed sessions $TF(U_i)$ and $TF(U_j)$ as

$$Sim(TF_u(U_i), TF_v(U_j)) = \mathrm{Min}(\overline{Sij}, \overline{Sji}), \quad (2)$$

where $\overline{Sij} = \mathrm{Avg}_{s=1,\ldots,n}\ (\mathrm{Max}_{t=1,\ldots,m}\{Sim(S_{is}, S_{jt})\})$ and $\overline{Sji} = \mathrm{Avg}_{t=1,\ldots,m}\ (\mathrm{Max}_{s=1,\ldots,n}\{Sim(S_{is}, S_{jt})\})$. Actually, $\overline{Sij}$ indicates the average degree that user i is similar to user j, while $\overline{Sji}$ is the average degree that user j is similar to user i, and $Sim(TF_u(U_i), TF_v(U_j))$ is the mutual similarity between the two framed navigation sessions.

3.2. Mining association rules

The purpose of mining association rules is to find out which web pages are usually visited together in a session. Operated on the clusters of time-framed navigation sessions, the association rules discovered for each cluster will characterize the navigation patterns of specific user groups. As a result, these clustered association rules can serve as the knowledge models to predict the next navigation requests for future similar users. To achieve this, a user classification method is needed to identify the cluster of navigation patterns to which the current user is most similar.

3.3. The user classification method

Recall that each cluster of timed-framed sessions has a medoid, which is a frame of navigation sessions from some user. The medoid in some sense represents a typical user navigation pattern for users from that cluster. For a specific user, the cluster to which the user is most similar can be selected by choosing the medoid to which the user's current behavior is most similar. Suppose there is a user k, and the user's most recent time frame consists of the previous sessions $S_1, S_2, \ldots, S_{n-1}$ and a current session S_n consisting of pages p_1 and p_2, as shown below

$U_k : S_1, S_2, \ldots, S_{n-1}, S_n[p_1, p_2 \Rightarrow ?].$

What will the user visit next after having visited pages p_1 and p_2? To answer this kind of questions, we select the cluster to whose medoid the user k is most similar. The computation is similar to that described in Section 3.1.2, except that in this case we do not need to calculate the degree a medoid is similar to the user. That is, while the user's current behavior may be very similar to (or part of) the selected medoid's, the medoid's may be of little similarity to the user's current behavior. Hence we do not consider the mutual similarity between a medoid and the user, as its value might be very low due to the incompleteness of the user's current frame sessions.

After classifying user k as a member of the cluster to which it is most similar, the association rules in the corresponding knowledge model of the cluster can be used to match the pages in the current session S_n of user k. Those rules matched with sufficient confidence (greater than a confidence threshold) will be fired, and the predicted items are added into the recommendation list in a sorted manner according to their decreasing confidence values. Furthermore, items that are suggested by more than one rules will be added to list only once with the highest confidence value.

On the other hand, to investigate the issue of whether it is better to consider the previous sessions $(S_1, S_2, \ldots, S_{n-1})$ when doing user classification, another avenue of user classification based on only the current session S_n will be investigated in Section 6 later.

4. Recommendation mechanisms

The recommendation mechanism is started right after a user has made his/her first request to a web site. However, it is quit often that the current session of the user matches no association rules at all. So, we need a recommendation mechanism that can provide reasonable suggestions when facing such a situation. In the following, we present two mechanisms for this purpose.

4.1. The window-sliding method

This method uses a sliding window technique to control the number of session pages to be matched against the association rules (Mobasher, Dai, Luo, & Nakagawa, 2001) Let $S_n = [p_1, p_2, \ldots, p_k]$ be the user's current session. Initially, the window covers all pages in S_n, and hence all pages $(p_1, p_2, \ldots, p_k)$ in the current session are used to match against the association rules. If no matched association rules could be found, the window would slide one position to the right, leaving the pages $p_2, \ldots, p_k$ for rule matching. While the sliding actions will lose more and more information about the user's navigation behavior, it does preserve the most recent information as possible as it can. The sliding process will repeat until at least one rule is matched or the window coverage becomes empty. For the latter case, we say that the user cannot receive the recommendation service under his/her current navigation session.

4.2. The maximal-matching method

In contrast to using a sliding window to preserve only the most recent session information for the matching work, the maximal-matching method preserves as much session information as possible for the matching work. This is achieved by finding all maximal subsets of the session pages that match successfully against the association rules. Given a set P of session pages, any subset M of P is called maximal if it matches at least one of the association rules, and no proper upper-set of M, which is also a subset of P, can find a matching rule. An efficient graph-based algorithm was implemented to find all the maximal-matching subsets of a page set given a set of association rules. To achieve this purpose, we use a lattice structure, as shown in Fig. 3, to store large itemsets discovered in the association-mining phase.

In Fig. 3, bold circles denote large itemsets, while dotted ones indicate itemsets that are not large (and actually are not stored). Numeric labels around the circles denote the supports of the corresponding itemsets. This lattice structure is useful for storing association rules as well as for finding maximal-matching itemsets. For example, the confidence of the association rule $p_a \rightarrow p_b$ can be easily computed by dividing the support of node $p_a p_b$ over that of node p_a. Besides, suppose the user has already accessed the web pages p_a, p_b, and p_c in his/her current session. Then the maximal-matching subsets of (p_a, p_b, p_c) can be found by traversing the graph starting from the corresponding page nodes (i.e. nodes of p_a, p_b and p_c), until reaching nodes that have no up-going edges to parent nodes that are subsets of (p_a, p_b, p_c). All the finally reached nodes form the set of

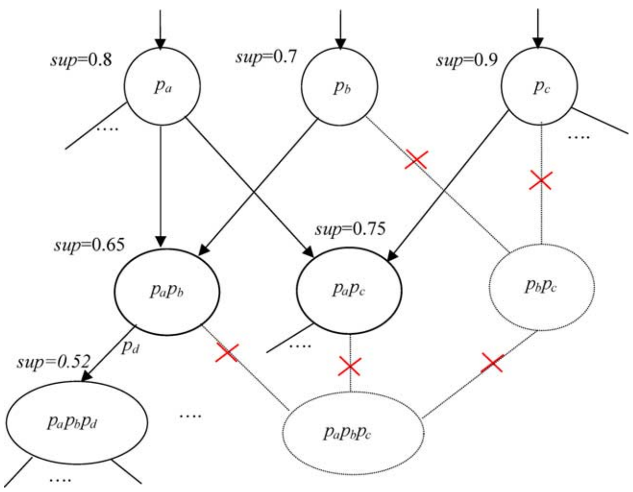

Fig. 3. A lattice graph storing large itemsets for association rule generation and finding maximal-matching subsets.

maximal-matching nodes (in this case, they are the $p_a p_b$ and $p_a p_c$). Each of the maximal-matching itemset is then used as the navigation information to match against the association rules. Actually, the rule matching work can also be done efficiently through the lattice of itemsets by locating the node with the itemset label and recommend the items corresponding to its out-going edges. For example, the maximal-matching itemset ($p_a p_b$) in Fig. 3 will recommend the page p_d with a confidence of 0.8 ($=0.52/0.65$). Again, if no maximal-matching itemsets could be found, we say that the user cannot receive the recommendation service under his/her current navigation session.

4.3. Performance criteria

Two measurements are often used to evaluate the effectiveness of the prediction knowledge obtained through the data analysis methods. One is the precision, and the other is the recall rate. The precision measures the system ability to provide correct predictions; while the recall rate measures the system ability to provide as many correct predictions as users need. A third criterion adopted in this paper is the coverage rate that measures the system ability to provide recommendation services for future unseen users. Specifically, traditional definitions of the three evaluation criteria are described as below.

1. Precision rate: the ratio of the recommended items that users actually need to the total items in the recommendation list.
2. Recall rate: the ratio of the recommended items that are needed by users to the total items that users actually need.
3. Coverage rate: the ratio of the users that receive the recommendation services.

The concepts of the evaluation criteria given above do not take practical navigational behavior into consideration. Actually, items recommended in top positions are more likely to be checked by users. Besides, it is always better for users' convenience sake to place items that users actually need in top positions of the recommendation list. Therefore, this research thinks differently about the item positions in the recommendation list. Accordingly, in this paper we adopt the weighted precision rate (Breese, Heckerman, & Kadie, 1998) and the recall rate to evaluate the effectiveness of recommendation. In the following we give the definitions of both weighted criteria.

4.3.1. Weighted precision rates

Let $A_i = (n_1, \ldots, n_{|Ai|})$ denote the set of items a user actually need during a navigation session i, and let $R_i = [r_1, \ldots, r_{|Ri|}]$ denote the ordered set of items that the system recommends to the user at some stage of the navigation session i. Let W_j be the weight of the top jth position in the recommendation list, which is defined as

$$W_j = \frac{1}{2^{(j-1)/(\alpha-1)}}, \tag{3}$$

where $|R_i|$ is the length of the recommendation list, and α is a parameter that specifies the item position where users have a 50–50 chance of viewing the item located. In this research we assume $\alpha = 10$, which indicates that the top 10-item positions have a probability of 0.5 and above for being explored by users (Breese et al., 1998). The weighted precision rate WP_i for the ith user session is then defined as

$$WP_i = \frac{\sum_{j=1,\ldots,|R_i|} H_j \times W_j}{\sum_{j=1,\ldots,|R_i|} W_j}, \qquad \begin{cases} 1, & \text{if hit,} \\ 0, & \text{otherwise,} \end{cases} \tag{4}$$

where $H_j = 1$ if the jth item of the recommendation list R_i is in the set of user needs A_i; otherwise it is 0. Besides, as the length of a recommendation list varies from session to session, we would like to know how best the system can do in precision for the ith session under the list length of N. The best situation happens when the items located continuously in front part of the list are what the user actually needs. So define

$$WP_i^{\max} = \frac{\sum_1^{\min(|A_i|,|R_i|)} W_j}{\sum_{j=1,\ldots,|R_i|} W_j}. \tag{5}$$

Therefore, a normalized average of weighted prediction rates for the total sessions is obtained by

$$AWP = \frac{\sum_{i=1}^{S} WP_i}{\sum_{i=1}^{S} WP_i^{\max}}, \tag{6}$$

where S is the total number of user sessions.

4.3.2. Weighted recall rates

The same formula (3) of position weight given above is applied. As conventional recall rate is the ratio of the recommended items that are needed by users to the total items that users actually need. A weighted version of the recall rate is then given as below

$$WR_i = \frac{\sum_{j=1,\ldots,|R_i|} H_j \times W_j}{\sum_{j=1,\ldots,|A_i|} W_j}, \qquad \begin{cases} 1, & \text{if hit,} \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

where $H_j = 1$ if the jth item of R_i is in A_i; otherwise it is 0. Besides, as the number of items that the user needs varies from session to session, we would like to know how best the system can do in recall rate for the ith session under the amount of the user's needs ($|A_i|$). The best situation happens when what the user actually needs are located continuously in front part of the list. So define

$$WR_i^{\max} = \frac{\sum_1^{\min(|A_i|,|R_i|)} W_j}{\sum_{j=1,\ldots,|A_i|} W_j}. \tag{8}$$

Therefore, a normalized average of weighted recall rates over the total sessions is obtained by

$$AWR = \frac{\sum_{i=1}^{S} WR_i}{\sum_{i=1}^{S} WR_i^{\max}}, \quad (9)$$

where S is the total number of user sessions.

5. An application

In this section, we present the application of the recommendation method to a web-based Virtual Classroom in Ming Chuan University (http://www.eduplanet.mcu.edu.tw/), in which the learning activities include the browsing of course syllabus, course material, learning sheet, and worksheet, online testing, group discussion, BBS, chatting room and so on. This educational web site is built to support the notion of managed knowledge space that facilitates the creation, sharing and exchanging of knowledge, where knowledge contributors could be teachers and/or students. The course material includes those produced by teachers as well as those knowledge documents created and organized by students through a cooperative learning process. As a result, it provides an opportunity for teachers and students to work together to enrich the knowledge space from diversity of perspectives.

5.1. Data collection and preprocessing

In this learning environment, students can choose and browse material according to the topic indices and perform further study following the hyperlinks embedded in the documents, or they can browse specific material through the system's search engine utility. All documents are displayed in a browsing window. A client agent is designed to track the user's activities, including the URLs of the pages showing in the browsing window, and sends them back to the behavior-tracking database on the server side.

Since the study focuses on the browsing related activities, all other unrelated log data are filtered out, including those activities of teachers as well as those browsing records with short staying time. In particular, browsing records with short staying time are often caused by pages that contain intermediate hyperlinks between web documents. For example, a student may intend to browse page-B, but have to browse page-A first because only through the hyperlink in page-A can he/she reaches page-B. In such a situation, page-A is often called a *pass-by page*. On the other hand, student's short references of pages may also be caused due to mistaking some pages as useful for their learning purposes. This kind of references can also be filtered out by checking a minimal page residence time (say 10 s).

Furthermore, the raw log data has to be reconfigured for further analysis. Table 2 shows the record format of the logged learning activities. All browsing records are sorted in an ascendant manner with the user id as a major key and a starting time tag as a minor key. Sessions are identified by packing continuous records that follow a 'login' type record until the next 'login' record. Specifically, browsing records picked up between two successive login-type records are grouped into a *browsing-session* record.

Table 2
The record format of logged learning activities

Data field	Description
Student id	Identifier of students
Page URL	URL of the referenced page
Activity Type	Activity type such as 'login', 'browsing', 'group discussion' and so on
Start Time	Start time of the activity
Stay Time	Staying time of the activity (in seconds)

6. Experiments and results

6.1. Design of the experiments

Several factors have impacts on the performance of the recommendation method. They include the length of the time frame, the user classification method, the recommendation policies, the confidence threshold of recommendation, and the amount of training data. Historical navigation data is collected from three classes (classes A, B and C) of a virtual classroom course ('Expert System') for one semester. These data will be preprocessed using the method described above. The sizes of the session databases for each class are listed in Table 3.

Three sets of experiments with different amounts of training data were conducted. The first one used a half of the session data from class A for training, and the other half for testing. The second one used the whole data of class A for training, and the whole data of class B for testing. The last experiment used the data of class A and B for training, and used the data of class C for testing. By comparing the results, we could investigate the issues of the impacts the amount of training data could have on the performance of the recommendation method.

For each set of experiments, we conduct the user-clustering experiments with a frame length of a week and a semester, respectively. The results will be compared to the non-clustering one reported in (Wang & Thao, 2003).

Table 3
Sizes of the session databases for each class

Class	Number of sessions	Number of users	Average session length
A	69	570	4.89
B	63	636	4.97
C	45	472	5.02

Table 4
Clustering results of the data sets under different frame sizes

Frame size	Class	Number of framed sessions	Number of clusters
Semester	Half of A	35	5
	A	69	7
	A + B	132	17
Week	Half of A	152	26
	A	315	29
	A + B	632	37

Furthermore, though practically the system can provide recommendation services at every stage of a user session, we just take the performance results of the recommendations provided at three typical stages of a user session: the first, middle and last ones. Recommendations provided at the first stage of a user session are given when there is only one item in the current session, and those provided at the middle stage are given when half the items in the user session are available. At last, recommendations provided at the last stage of a user session are given when all but one item in the current session is available. The total service performance provided for a user session is then computed as an average of the performance results at the three stages of the session.

At last, some parameter settings are given here. The support threshold applied in association mining is 0.02. The confidence thresholds of recommendation are 0.1, 0.2 and 0.3, respectively. The parameter α for computing position weights is 10, and the minimal intra-cluster similarity is 0.3.

Table 4 shows the clustering results based on the semester and week frame sizes, respectively.

6.2. Results and discussion

Let AWP denote the average weighted precision rate, AWR be the average weighted recall rate, and ASR be the average coverage rate Besides, P denotes the recommendation policy, which takes the value of either MM (maximal matching) or WS (window sliding). CF denotes the confidence threshold.

Table 5 shows the results of the experiments that use half the navigation data from class A for training, and the other half for testing. Table 6 shows the results of the experiments that use the whole data of class A for training, and the whole data of class B for testing. Table 7 shows the results of the experiments that use the data of class A and B for training, and the data of class C for testing. Table 8 shows the results of the non-clustering experiments (Wang & Thao, 2003).

Through the above experiments, we will discuss the issues listed below.

1. Is the method with clustering based on time-framed navigation sessions better than the non-clustering one?
2. Is the method with shorter time frames better than the one with larger ones?
3. Is the method with maximal-matching policy better than the one with window-sliding policy?
4. Is the method with user classification scheme considering a user's previous sessions better than the one considering only the current session?

Table 5
Results of the experiment that uses half the navigation data from class A for training, and the other half for testing (half A → half A)

CF	Time frame—semester											
	User classification—considering the previous sessions in a time frame						User classification—considering only the current session					
	P						*P*					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.47	0.52	0.61	0.38	0.47	0.61	0.45	0.49	0.57	0.37	0.44	0.57
CF = 0.2	0.46	0.50	0.61	0.38	0.45	0.61	0.43	0.46	0.57	0.37	0.43	0.57
CF = 0.3	0.41	0.45	0.61	0.35	0.4	0.61	0.39	0.43	0.57	0.33	0.39	0.57
CF	Time frame—week											
	User classification—considering the previous sessions in a time frame						User classification—considering only the current session					
	P						*P*					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.55	0.64	0.44	0.54	0.63	0.44	0.58	0.64	0.44	0.56	0.64	0.44
CF = 0.2	0.55	0.62	0.44	0.53	0.60	0.44	0.57	0.64	0.44	0.56	0.63	0.44
CF = 0.3	0.53	0.57	0.44	0.52	0.57	0.44	0.56	0.60	0.44	0.55	0.60	0.44

Table 6
Results of the experiments that use the whole data of class A for training, and the whole data of class B for testing (A → B)

CF	Time frame—semester											
	User classification—considering the previous sessions in a time frame						User classification—considering only the current session					
	P						P					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.49	0.54	0.59	0.42	0.49	0.59	0.46	0.51	0.57	0.37	0.46	0.57
CF = 0.2	0.46	0.51	0.59	0.39	0.45	0.59	0.41	0.46	0.57	0.35	0.42	0.57
CF = 0.3	0.37	0.43	0.59	0.32	0.38	0.59	0.36	0.43	0.57	0.31	0.38	0.57
CF	Time frame—week											
	User classification—considering the previous sessions in a time frame						User classification—considering only the current session					
	P						P					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.57	0.62	0.52	0.51	0.59	0.52	0.58	0.65	0.51	0.53	0.61	0.51
CF = 0.2	0.50	0.59	0.52	0.45	0.55	0.52	0.50	0.60	0.51	0.46	0.57	0.51
CF = 0.3	0.47	0.55	0.52	0.43	0.51	0.52	0.47	0.56	0.51	0.43	0.53	0.51

5. How does the confidence threshold affect the performance of the recommendation method?
6. Is the method trained with a larger amount of data better than the one trained with smaller amount of data?

6.2.1. Is the method with clustering based on time-framed sessions better than the non-clustering one?

Table 9 shows that the clustering recommendation methods based on the 'week' frame size have better results

Table 7
Results of the experiments that use the data of class A and B for training, and the data of class C for testing (A, B → C)

CF	Time frame—semester											
	User classification—considering the previous sessions in a time frame						User classification—considering only the current session					
	P						P					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.43	0.53	0.59	0.35	0.47	0.59	0.52	0.59	0.52	0.44	0.54	0.52
CF = 0.2	0.40	0.48	0.59	0.34	0.43	0.59	0.51	0.55	0.52	0.43	0.52	0.52
CF = 0.3	0.34	0.44	0.59	0.29	0.39	0.59	0.46	0.5	0.52	0.40	0.47	0.52
CF	Time frame—week											
	User classification—considering the previous sessions in a time frame						User classification—considering only the current session					
	P						P					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.55	0.65	0.51	0.46	0.61	0.51	0.54	0.65	0.50	0.48	0.62	0.50
CF = 0.2	0.46	0.57	0.51	0.40	0.53	0.51	0.47	0.57	0.50	0.42	0.54	0.50
CF = 0.3	0.45	0.53	0.51	0.39	0.49	0.51	0.44	0.53	0.50	0.40	0.50	0.50

Table 8
Results of the non-clustering experiments (Wang & Thao, 2003)

CF	Non-clustering (half A → half A)						Non-clustering (A → B)					
	P						*P*					
	MM			WS			MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.46	0.52	0.63	0.38	0.47	0.63	0.45	0.50	0.64	0.4	0.46	0.64
CF = 0.2	0.44	0.49	0.63	0.36	0.45	0.63	0.43	0.47	0.64	0.37	0.43	0.64
CF = 0.3	0.39	0.46	0.63	0.34	0.41	0.63	0.36	0.41	0.64	0.31	0.37	0.64

CF	Non-clustering (A, B → C)					
	P					
	MM			WS		
	AWP	AWR	ASR	AWP	AWR	ASR
CF = 0.1	0.42	0.53	0.62	0.32	0.46	0.62
CF = 0.2	0.41	0.50	0.62	0.32	0.42	0.62
CF = 0.3	0.32	0.43	0.62	0.25	0.35	0.62

in regards to both the weighted prediction and recall rates than the non-clustering ones by more than 10%. However, in the mean time they have lower coverage rates than the non-clustering ones by more than 10%, too. Hence, there appears an obvious tradeoff among the three service quality measures (precision, recall and coverage). Furthermore, the recommendation methods with clustering based on a semester frame size are comparable with the non-clustering

Table 9
Performance results sorted by decreasing AWP for each data set with CF = 0.1

Class data	Cluster	Time frame	User classification	Policy	CF	AWP	AWR	ASR
Half A → halfA	Yes	Week	Current	MM	0.1	0.58	0.64	0.44
Half A → halfA	Yes	Week	Current	WS	0.1	0.56	0.64	0.44
Half A → halfA	Yes	Week	Prev + current	MM	0.1	0.55	0.64	0.44
Half A → halfA	Yes	Week	Prev + current	WS	0.1	0.54	0.63	0.44
Half A → halfA	Yes	Semester	Prev + current	MM	0.1	0.47	0.52	0.61
Half A → halfA	No	×	×	MM	×	0.46	0.52	0.63
Half A → halfA	Yes	Semester	Current	MM	0.1	0.45	0.49	0.57
Half A → halfA	Yes	Semester	Prev + current	WS	0.1	0.38	0.47	0.61
Half A → halfA	No	×	×	WS	×	0.38	0.47	0.63
Half A → halfA	Yes	Semester	Current	WS	0.1	0.37	0.44	0.57
A → B	Yes	Week	Current	MM	0.1	0.58	0.65	0.51
A → B	Yes	Week	Prev + current	MM	0.1	0.57	0.62	0.52
A → B	Yes	Week	Current	WS	0.1	0.53	0.61	0.51
A → B	Yes	Week	Prev + current	WS	0.1	0.51	0.59	0.52
A → B	Yes	Semester	Prev + current	MM	0.1	0.49	0.54	0.59
A → B	Yes	Semester	Current	MM	0.1	0.46	0.51	0.57
A → B	No	×	×	MM	×	0.45	0.5	0.64
A → B	Yes	Semester	Prev + current	WS	0.1	0.42	0.49	0.59
A → B	No	×	×	WS	×	0.4	0.46	0.64
A → B	Yes	Semester	Current	WS	0.1	0.37	0.46	0.57
A, B → C	Yes	Week	Prev + current	MM	0.1	0.55	0.65	0.51
A, B → C	Yes	Week	Current	MM	0.1	0.54	0.65	0.5
A, B → C	Yes	Semester	Current	MM	0.1	0.52	0.59	0.52
A, B → C	Yes	Week	Current	WS	0.1	0.48	0.62	0.5
A, B → C	Yes	Week	Prev + current	WS	0.1	0.46	0.61	0.51
A, B → C	Yes	Semester	Current	WS	0.1	0.44	0.54	0.52
A, B → C	Yes	Semester	Prev + current	MM	0.1	0.43	0.53	0.59
A, B → C	No	×	×	MM	×	0.42	0.53	0.62
A, B → C	Yes	Semester	Prev + current	WS	0.1	0.35	0.47	0.59
A, B → C	No	×	×	WS	×	0.32	0.46	0.62

ones. The results show that the clustering recommendation methods based on the 'week' frame size can characterize the users' behavior more accurately than those with a 'semester' size as well as those non-clustering ones, so resulting in higher precision and recall rates. However, both the clustering methods do not cover the access patterns of the testing data (though increasing the size of the training data helps) as well as the non-clustering methods do. Nevertheless, if we apply AWP × ASR and AWR × ASR as the expected service performances of prediction and recall rates, respectively, the methods with time-framed clustering beat the non-clustering ones in moderate and large training sets.

6.2.2. Is the method with shorter time frames better than the one with larger ones?

From Table 9, it is obvious that the clustering methods with the frame period of a week are almost better in precision and recall rates than those with a semester by about 10%, but in the mean time are worse in coverage rate by about 10%. This might be because of the scope of week clustering contains fewer sessions than the semester one, and hence it can find fewer association rules to provide services for various kinds of users.

The results show that the clustering recommendation methods based on the 'week' frame size can characterize the users' behavior more accurately than those with a 'semester' size. However, the clustering methods with the semester frame cover the access patterns of the testing data better than the methods with week frame, hence providing higher coverage rates.

6.2.3. Is the method with maximal-matching policy better than the one with window-sliding policy?

From Table 9, it is found that, when other parameters are fixed, the methods with maximal matching are always better in precision and recall rates than those with window-sliding policy. It is due to the fact that the maximal-matching method finds more useful associations rules than the window-sliding ones. Nevertheless, it has nothing to do with the coverage rate whether the policy is maximal matching or window sliding. The results show the maximal-matching technique does help improve the performance of the recommendation system by near 10%.

6.2.4. Is the method with user classification scheme considering a user's previous framed sessions better than the one considering only the current session?

A fact from Table 9 is that the methods with user classification scheme considering a user's previous framed sessions are always better in the coverage rates than those with the one considering only the current session. This is significantly true for the methods with semester time frame when the data set is large (better by 7%). It shows the classification scheme combing current sessions with previous framed sessions helps in improving the coverage rate, especially for large frame sizes.

It is also found that, when the data sets are small and moderate and the frame size is semester, the methods with user classification scheme considering a user's previous framed sessions are better in precision and recall rates than those considering only the current session. However, when the frame size is week, the situation goes counter to when it is semester. That is, when the data sets are small and moderate, the methods with longer frame size (semester) should adopt the user classification scheme combing previous frame sessions and current sessions, while the methods with shorter frame size (week) should adopt the user classification scheme considering only the current sessions (though the improvement is only 1–3%).

On the other hand, as the dataset becomes large, it is interesting to find that the methods with the user classification scheme considering only the current session are almost better in precision and recall rates than those combining a user's previous framed sessions and the current session. This is significantly true for the methods with semester time frame (better by 9 and 6% in precision and recall rates, respectively). The results show that in large data set, the current sessions play more dominant roles in classifying users to proper clusters for either semester or week frame size.

6.2.5. How does the confidence threshold affect the performance of the recommendation method?

It is obviously found from Tables 5–8 that both the precision and recall rates decrease as the confidence threshold increases for both the clustering and non-clustering methods. This might be due to the fact that it is harder to find applicable rules when the confidence threshold is high. Besides, it can also be found that the confidence threshold has nothing to do with the coverage rates. The results show that a sufficiently lower confidence threshold will improve both the precision and recall rates.

6.2.6. Is the method trained with a larger amount of data better than the one trained with smaller amount of data?

It is found that the precision and recall rates decrease when the size of the dataset increases for non-clustering methods. It shows that the non-clustering method does not do well with the increases of navigation patterns caused by increasing the data size. On the other hand, it is found that the precision rate increases when the training data size increases from half a class to one class. However, the precision rate decreases as the training data size increases from one class to two classes. In another experiment with a lowered support threshold for mining association rules, the precision rate rises back to a comparable position. This suggests that the recommendation system might adopt a lower support threshold appropriately to discover useful association rules for large data set.

7. Conclusive remarks

This research proposes a personalized recommendation based on time-framed navigation clustering and data mining technology. The experimental results show that the best average weighed precision rate is 0.6, average weighted recall rate is 0.7 and average service rate is 0.5, respectively. It shows that our method is better in precision and recall rates than the conventional non-clustering one, and is comparable in the service coverage rate. The results also suggest that the recommendation method uses a shorter frame size such as a week for clustering user navigations and mining association rules, because a shorter frame size could track more flexibly the changes of users' traversal behavior. As to the recommendation policies, the results show that our maximal-matching policy is significantly better than the window-sliding one.

However, the experimental results are preliminary and conservative because of the size of data sets used in the experiments are of moderate scale. Larger scaled experiments will be conducted to further confirm the effectiveness of the method. Besides, one limitation of this method is the inherent problem caused by the low supports of web page navigations, making it harder to build appropriate knowledge models. This is often solved by choosing a properly low-support values used to mine the association rules, as is adopted in this research. Other association-mining techniques (Wang, He, & Han, 2003) could be applied to avoid the low-support problem. Another future work is to probe the effect of the knowledge model built by combining framed session clustering with mining sequential patterns.

References

Agrawal, R., Imielinski, T., & Swami, A. (1993). *Mining association rules between sets of items in large databases*. *Proceedings of ACM SIGMOD* (pp. 207–216).

Agrawal, R., & Srikant, R. (1994). *Fast algorithm for mining association rules*. *Proceedings of the VLDB conference* (pp. 487–499).

Breese, J. S., Heckerman, D., & Kadie, C. (1998). *Empirical analysis of predictive algorithms for collaborative filtering*. *Proceedings of the fourteenth conference on uncertainty in artificial intelligence* (pp. 43–52).

Chen, M. S., Han, J., & Yu, P. S. (1996). Data mining: an overview from a database perspective. *IEEE Transactions on Knowledge and Data Engineering*, *8*(6), 866–883.

Cooley, R., Mobasher, B., & Srivastava, J. (1997). *Web mining: Information and pattern discovery on the World Wide Web*. *Proceedings of IEEE international conference tools with AI* (pp. 558–567).

Eirinaki, M., & Vazirgiannis, M. (2003). Web mining for Web personalization. *ACM Transactions on Internet Technology*, *3*(1), 1–27.

Fu, X., Budzik, J., & Hammond, K. J (2000). Mining navigation history for recommendation. In *Proceedings of the fifth international conference on intelligent user interfaces* (pp. 106–112).

Gery, M., & Haddad, H. (2003). *Evaluation of Web usage mining approaches for user's next request prediction*. *Proceedings of the fifth ACM international workshop on Web information and data management* (pp. 74–81).

Han, J., & Kamber, M. (2001). *Data mining, concepts and techniques: Cluster analysis,* Chap.8. *San Francisco: Morgan Kaufmann.*

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Reial, J. (1997). GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, *40*(3), 77–87.

Kwak, M., & Cho, D. S. (2001). *Collaborative filtering with automatic rating for recommendation*. *Proceedings of ISIE 2001 IEEE international symposium on industrial electronics* (pp. 625–628).

Lee, C. H., Kim, Y. H., & Rhee, P. K. (2001). Web personalization expert with combining collaborative filtering and association rule mining technique. *Expert Systems with Applications*, *21*, 131–137.

Mobasher, B., Cooley, R., & Srivastava, J. (2000). Automatic personalization based on Web usage mining. *Communications of the ACM*, *43*(8), 142–151.

Mobasher, B., Dai, H., Luo, T., & Nakagawa, M. (2001). *Effective personalization based on association rule discovery from Web usage data*. *Proceedings of the third international workshop on web information and data management* (pp. 9–15).

Mulvenna, M. D., Anand, S. S., & Buchner, A. G. (2000). Personalization on the net using Web mining. *Communications of the ACM*, *43*(8), 123–125.

Nichols, D. M. (1997). *Implicit rating and filtering*. *Proceedings of the fifth workshop on filtering and collaborative filtering* (pp. 31–36).

Riecken, D. (2000). Personalized views of personalization. *Communications of the ACM*, *43*(8), 27–28.

Wang, F. H., & Thao, S. M. (2003). *A study on personalized Web browsing recommendation based on data mining and collaborative filtering technology*. *Proceedings of national computer symposium, Taiwan* (pp. 18–25).

Wang, K., He, Y., & Han, J. (2003). Pushing support constraints into association rules mining. *IEEE Transactions on Knowledge and Data Engineering*, *15*(3), 642–658.