American Journal of Applied Sciences 8 (3): 277-283, 2011
ISSN 1546-9239
© 2010 Science Publications

IPACT: Improved Web Page Recommendation System Using Profile Aggregation Based On Clustering of Transactions

Yahya AlMurtadha, Md. Nasir Bin Sulaiman, Norwati Mustapha and Nur Izura Udzir
Department of Computer Science, Faculty of Computer Science and Information Technology,
University Putra Malaysia, 43400 UPM Serdang, Selangor Darul Ehsan, Malaysia

Abstract: Problem statement: Recently, Web usage mining techniques have been widely used to build recommendation systems especially for anonymous users. **Approach:** Assigning the current user to the best web navigation profile with similar navigation activities will improve the ability of the prediction engine to produce a recommendation list then introduce it to the user. This study presents iPACT an improved recommendation system using Profile Aggregation based on Clustering of Transactions (PACT). **Results:** iPACT shows better prediction accuracy than the previous methods PACT and Hypergraph. **Conclusion:** The users' interests change over time; hence an incremental and adaptive web navigation profiling is a key feature for the future works.

Key words: Recommender systems, web usage mining, web navigation profiles, prediction engine, profile aggregation, Collaborative filtering (CF), Case-Based Reasoning Plan Recognition (CBRPR), hybrid filtering, Web Usage Mining (WUM), Longest common Sequences algorithm (LCS)

INTRODUCTION

Web Recommendation system is a specific type of information filtering system technique that attempts to predict the user next browsing activity then recommend to the user web pages items that are likely to be of interest to the user. A recommender system is a typical software solution used in e-commerce for personalized services. Based on the customer preferences, It helps to find the products they would like to purchase by providing recommendations and is particularly useful in e-commerce sites that offer millions of products for sale (Kim *et al*., 2005). According to (Gao *et al*., 2009) there are four filtering approaches for making recommendations, namely, rule-based filtering, content-based filtering, collaborative filtering and hybrid filtering.

Rule-based recommendation: Rule-based filtering approach is based on "if this, then that" rules processing. The primary drawback of rule-based filtering techniques is the bias caused by the subjective description of users or their interests by the users themselves as input.

Content-based recommendation: this kind of recommendation system is Based on a comparison between items and users profiles (Park and Chang, 2009). Suhasini *et al*. (2008) is an examples of image retrieval content-based filtering systems include. Content analysis is practical only if the items have well-defined attributes and those are attributes can be extracted automatically; for some multimedia, such as audio/video stream and graphical images, the content analysis is hard to apply.

Collaborative Filtering (CF) based recommendation: CF system recommends products based on the similarity of the preferences of a group of customers known as a neighbor (Kim *et al*., 2005). CF suffers from the cold start problem since it usually provides very bad predictions/recommendations to new users having very few collections. Other significant limitations are the high computation cost that goes linear with the increase number of users and items and the sparsity of the dataset. Similarity indexing, dimensionality reduction and offline clustering have been proposed to remedy these weaknesses (Gao *et al*., 2009).

Hybrid filtering based recommendation: Such recommendation aims to avoid certain limitations of filtering methods by combining two or more filtering methods together. For example, (Barragáns-Martínez *et*

Corresponding Author: Computer Science department, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 UPM Serdang, Selangor Darul Ehsan, Malaysia
Tel: +60-3-8946-6585, Fax: +60-3-8946-6577

al., 2010) described the design, development and startup of queveo.tv: a Web 2.0 TV program recommendation system with a hybrid approach (which combines content filtering techniques with those based on collaborative filtering.

Web usage mining: Based on the data used in the mining process, Web mining tasks can be categorized into three main types: Web content mining, Web structure mining and Web usage mining. Web content mining extracts useful information/knowledge from Web page data such as text and graphics. Web structure mining discovers knowledge from hyperlinks, which represent the structure of the Web. (Kumar and Singh, 2010) presented a study on hyperlink analysis and the algorithms used for link analysis in the Web Information retrieval. Web Usage Mining (WUM) mines the data that describes the pattern of usage of Web pages, such as IP addresses, page references and the date and time of accesses. WUM run any number of data mining algorithms on usage data in order to analyze and then discover useful patterns in the navigational behavior of users. These patterns are discovered by applying some clustering algorithms on the preprocessor phase of the web usage mining and classification algorithms on the web mining process. The discovered patterns are usually represented as collections of pages, objects, or resources that are frequently accessed by groups of users with common needs or interests. The output of the WUM is some patterns that may be the input to the Recommendation systems Engine which is one of the application areas of the Web usage and gives the ability to predict the next visited page for a given user. Kusrini *et al*. (2010) developed CUC-C4.5 framework that had been applied on the case of differential diagnosis knowledge building in a group decision support system that might be used for recommendation systems. The various recommender models and analysis of the key features of those models and the features of portal sites that employ recommender systems to help the research community are addressed by (Ganapathy and Arunesh, 2010).

Previous works on WUM: Recently, Web usage mining techniques have been widely used to build recommendation systems. Various web usage mining techniques have been applied. Ramadhan *et al*. (2005) provided an updated focused survey of major aspects and problems related to the task of modeling the user behavior. Vijayalakshmi and Mohan, (2010) introduced an efficient strategy for discovering Web usage mining by using the application of sequential pattern mining techniques to discover usage patterns from Web data. Vijayalakshmi and Mohan, (2010) discussed how to maintain discovered sequential patterns when some information is deleted from a sequence database. Ren and Zhou, (2006) approach resulted in the generation of usage profiles and automatic identification of user interest in each profile. Sumathi *et al*. (2010) developed an application of session based clustering to analyze web pages of user interest from web log files. Dimopoulos *et al*. (2010) consider the problem of web page usage prediction in a web site by modeling users' navigation history and web page content with weighted suffix trees. Jalali *et al*. (2010) developed a recommendation system called WebPUM, an online prediction using Longest common Sequences algorithm (LCS) for classifying user navigation patterns to predict users' future intentions. AlMurtadha *et al*. (2010) proposed a method for Learning and mining the web navigation profiles to provide an appropriate model to recommend to the anonymous user. Göksedef and Gündüz-Ögüdücü (2010) investigated a hybrid recommender system, which combines the results of several recommender techniques based on web usage mining. Forsati and Meybodi (2010) proposed hybrid algorithm for web page recommendation distributed learning automata and weighted association rule mining. Castellano *et al*. (2011) presented NEWER as a usage-based Web recommendation system that exploits the potential of Computational Intelligence techniques to dynamically suggest interesting pages to users according to their preferences.

MATERIALS AND METHODS

iPACT recommendation system: The main purpose of this study is to improve the web page recommendation accuracy by improving the classification part of the recommendation engine. As shown in Fig. 1, the proposed recommendation architecture consists of two main phases, namely the offline and online. In the offline phase is responsible for partition the filtered sessionized transactions into clusters of similar pageviews. Then, generate the web navigation profiles based on these clusters of transactions using PACT methodology. The online phase is responsible for matching the new user transaction (current user session) to the profile shares common interests to the user. The proposed recommendation system is called iPACT. Fig. 2 shows the inputs and ouputs of each pahse of iPACT. The input to the offline phase is the preprocessed web server logs file and the outputs are 1) clusters of navigation transactions and 2) the web navigation profiles. The inputs to the online phase are 1) the web navigation profiles generated from the offline phase and

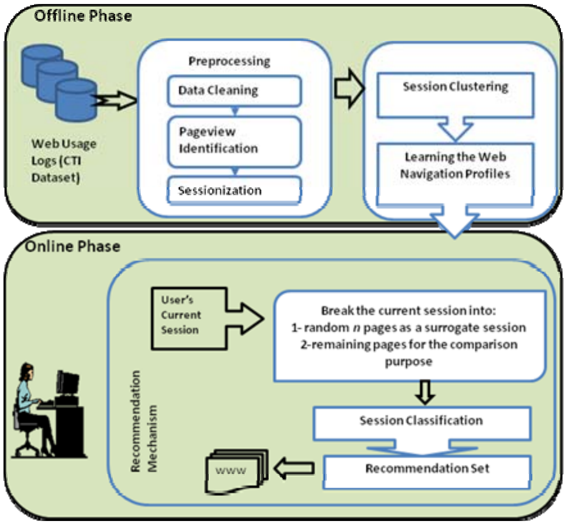
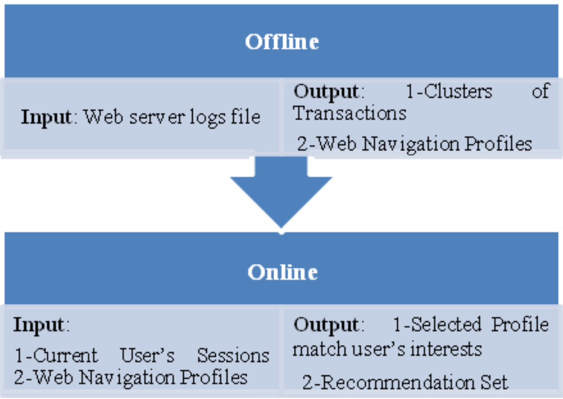


Fig. 1: iPACT Recommender System Architecture



Offline	
Input: Web server logs file	**Output**: 1-Clusters of Transactions 2-Web Navigation Profiles

Online	
Input: 1-Current User's Sessions 2-Web Navigation Profiles	**Output**: 1-Selected Profile match user's interests 2-Recommendation Set

Fig. 2: iPACT input and output

the current user session and the output will be the recommednation set in addition to the best profile that match the user interests. The following subsection will describe both phases in details.

The offline component: This section will describe the two processes taken by the offline phases, namely clustering of transactions and generating the web navigation profiles.

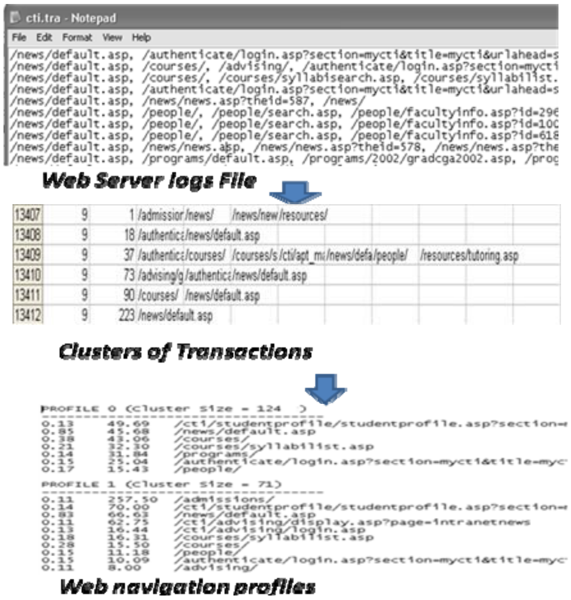
Clustering of transactions: Clustering is aimed at finding groups which share common navigation behaviors web usage mining is an important step. We used K-Mean clustering algorithm to cluster the preprocessed and filters web server logs with different K values. For the clustering purpose, we used CTI.std file as an input to the K-Means clustering algorithm. The file contains 13745 sessions with 682 pageviews visited by different users. The file represents a session-pageview matrix where each column is a pageview and each row is a session represented as a vector. The entries in the table correspond to the amount of time (in seconds) spent on pageviews during a given session. The pageview durations were maxed out at 999 seconds. For each session, the pageview duration of the last pageview in that session, was estimated to be the average duration of that pageview across all sessions (in which the pageview does not occur as the exit page). The output is K-clusters each contain navigation transactions with similar pageviews. These clusters are used later to generate web navigation profiles as will be explained in the following subsection.

Generating the web navigation profiles: The critical step is the effective derivation of good quality and useful navigation profiles from these patterns. Well partitioning of groups of anonymous users is critical to assign the anonymous user to the best group shares similar interests. Using only the clusters of transactions produced by the previous step is not effective since there is a strong need to filter out some pageviews with low navigation importance. Hence, the clusters produced by the clustering step (previous step) are used to generate the navigation profile with one profile for each cluster by setting the *min_sup* and *min_weight*. The web navigation profile contains only those pageviews that passed certain confidence support and weights values. The confidence support determines the frequency occurrence on those pages in the cluster. *min_sup* values are used to filter out profile elements which do not have sufficient support while *min_weight* values are used to filter out profile elements which have low average weight (navigation time spent visiting this page). To summarize we construct a web navigation profile as a set of pageview-weight pairs:

$$\text{profile}=\{\ p,\text{weight}(p)|\ p \in P,\ \text{weight}(p) \geq \text{min\_weight}\ \}$$

where P ={p1, p2, . . . , pn}, a set of n pageviews appearing in the transaction file with each pageview uniquely represented by its associated URL and the weight(p) is the (mean) value of the attribute's weights in the cluster. Fig. 3 shows the process of producing the web navigation profile.

The online component*:* After the navigation profiles are extracted from the previous sessions, many preprocessing steps are to be taken. First, all the profiles' pageviews are sorted in descending order according to their weights. Then, all the highly frequent



```
/news/default.asp, /authenticate/login.asp?section=mycti&title=mycti&urlahead=s
/news/default.asp, /courses/, /advising/, /authenticate/login.asp?section=mycti
/news/default.asp, /courses/, /courses/syllabisearch.asp, /courses/syllabilist.
/news/default.asp, /authenticate/login.asp?section=mycti&title=mycti&urlahead=s
/news/default.asp, /news/news.asp?theid=587, /news/
/news/default.asp, /people/, /people/search.asp, /people/facultyinfo.asp?id=296
/news/default.asp, /people/, /people/search.asp, /people/facultyinfo.asp?id=100
/news/default.asp, /people/, /people/search.asp, /people/facultyinfo.asp?id=618
/news/default.asp, /news/news.asp, /news/news.asp?theid=578, /news/news.asp?the
/news/default.asp, /programs/default.asp, /programs/2002/gradcga2002.asp, /prog
```

13407	9	1	/admissior	/news/	/news/new	/resources/			
13408	9	18	/authentic:	/news/default.asp					
13409	9	37	/authentic:	/courses/	/courses/s	/cti/apt_m:	/news/defa	/people/	/resources/tutoring.asp
13410	9	73	/advising/g	/authentic:	/news/default.asp				
13411	9	90	/courses/	/news/default.asp					
13412	9	223	/news/default.asp						

```
PROFILE 0 (Cluster Size = 124  )
------------------------------------
0.13     49.69    /cti/studentprofile/studentprofile.asp?section=
0.85     45.68    /news/default.asp
0.38     43.06    /courses/
0.21     32.30    /courses/syllabilist.asp
0.14     31.84    /programs/
0.15     25.04    /authenticate/login.asp?section=mycti&title=myc
0.17     15.43    /people/

PROFILE 1 (Cluster Size = 71)
------------------------------------
0.11     257.50   /admissions/
0.14     70.00    /cti/studentprofile/studentprofile.asp?section=
0.83     66.63    /news/default.asp
0.11     62.75    /cti/advising/display.asp?page=intranetnews
0.13     16.44    /cti/advising/login.asp
0.18     16.31    /courses/syllabilist.asp
0.28     15.50    /courses/
0.15     11.18    /people/
0.15     10.09    /authenticate/login.asp?section=mycti&title=myc
0.11     8.00     /advising/
```

Fig. 3: Generating of Web Navigation Profiles

pageviews like the index pages are removed. The online phase is the main component of the recommender system. It includes the prediction engine responsible for assigning the current user to the best web navigation profile and then predicts his future navigation activity.

The current user is matched against the web navigation profiles generated by the offline component. The best web navigation profile is chosen to be the source of prediction where by a recommendation list is to be created from those pageviews not visited by the user and attached to the user navigation list. Two sequences methods are applied to assure the classification. First, a statistical classification to assign the active session to the best web navigation profile with the highest number of matched pageviews. Second, use the cosine coefficient to find the similarity with the profiles that may meet or be missed using the first method. Finally, make a recommendation list based on these selected profiles from those pageviews that pass a certain threshold.

Since both the active session and the choose profile can be represented as vectors; the cosine coefficient commonly used in information retrieval was used to do the matching purpose:

$$\mathrm{profileMatch} = \frac{\sum_{i} w_i^c \cdot p_i^p}{\sqrt{\sum_{i} \left(w_i^c\right)^2 \times \sum_{i} \left(p_i^p\right)^2}} \quad (1)$$

Where w_i^c is the associate weight for the corresponding pageview reference in the active session in binary values (0 for absence and 1 for presence) and p_i^p is the associate weight for the corresponding pageview reference in the profile. A recommendation score is computed for those items not already visited by the user in the active session in order to recommend them based on their scores.

$$\mathrm{Re}\,cScore = \sqrt{\left[PageWeight * \mathrm{Pr}\,ofileMatch\right]} \quad (2)$$

If the recommendation score is higher than the recommendation threshold, then select it. Various values from 0.1 to 1.0 are taken for the recommendation threshold.

Experimental setup:

CTI dataset: Our experiments have been conducted on DePaul University CTI logs file dataset which contains the preprocessed and filtered sessionized data for the main DePaul CTI Web server (http://www.cs.depaul.edu). The data is based on a random sample of users visiting this site for a two-week period during April 2002. The original (unfiltered) data contained a total of 20950 sessions from 5446 users. The filtered data files were produced by filtering low support pageviews and eliminating sessions of size 1. The filtered data contains 13745 sessions and 683 pageviews. Based on the proposed architecture, a recommendation system is developed using Microsoft VC++ connected to Microsoft Access database through an Open Database Connection (ODBC).

Experimental evaluation: We used the precision, coverage and F1 standard measures in order to evaluate the recommendation effectiveness. Assume that we have active current session *A* taken from the evaluation set and we have *R* as a recommendation set using the prediction engine over the navigation profiles. *w* represents the items that have already been visited by the user in *A*. Precision measures the number of correct relevant recommendation to the total recommendations. The precision is defined as:

$$precision(R, A) = \frac{\left|R \cap (A - w)\ \right|}{\left|R\ \right|}$$

Coverage is the ratio between the number of relevant Web pages retrieved and the total number of web pages that actually belongs to the user session. The coverage measure is defined as:

$$Coverage(R, A) = \frac{\left|R \cap (A - w)\right|}{(A - w)}$$

The harmonic mean for both precision and coverage is used and defined as F1 measure:

$$F1(R,A)=\frac{2\times \text{precision}(R,A)\times \text{Coverage}(R,A)}{\text{precision}(R,A)+\text{Coverage}(R,A)}$$

The F1 measure attains its maximum value when both accuracy and coverage are maximized. Finally, for a given prediction thresholds, the mean overall sessions in the evaluation set is computed as the overall evaluation score for each measure.

RESULTS

This section will present several measures for evaluating the recommendation accuracy namely, precision, coverage and F1 and discuss the experimental results based on these measures.

The aim of the experiments is to evaluate the ability of iPACT on predicting the new visiting page to the current navigation session for testing the prediction accuracy of the proposed enhanced classification algorithm. Using CTI dataset, the basic methodology used as proposed by PACT is as follows. Each session in the dataset is divided into two parts: surrogate session with sliding window size *n* (in the experiments we used *n*=3) and the remaining for the comparison purpose. For a given transaction *t* in the evaluation set and an active session window size *n*, we randomly chose |t|−*n* +1 groups of items from the transaction as the surrogate active session windows, each having size *n*. For each of these active sessions, produced a recommendation set based on aggregate profiles and compared the set to the remaining items in the transaction (i.e., t −w) in order to compute the precision, coverage, F1 and R scores. For each of these measures, the final score for the transaction t was the mean score over all of the |t|−*n* +1 surrogate active sessions associated with *t*. Finally, the mean over all transactions in the evaluation set was computed as the overall evaluation score for each measure.

Table I shows the recommendation set contain pages recommended from the profile No.2 to the session with the following pages *(/news/default.asp, /courses/,/courses/syllabisearch.asp,/courses/syllabilist. asp, /people/facultyinfo.asp?id=231*) each with a recommendation score. That's mean the recommendation systems has chooses the profile No.3 (programs and advising) as the best source for predicting the next visited pages since this session contains pages pertaining courses.

Table 2 depicts the precision, coverage and F1 performance measurements for the proposed iPACT with various recommendation thresholds for the recommendation score. The table shows that the best recommendation accuracy obtained with recommendation score threshold value of 0.5.

Table 1: Example of recommendation Set for anonymous user's session

Recommended profile	Recommendation set	Recommendation score
2	/admissions/	2.269361
	/cti/advising/display.asp?page=intranetnews	1.120268
	/cti/advising/login.asp	0.573411
	/courses/syllabilist.asp	0.571139
	/courses/	0.556776
	/people/	0.472864

Table 2: Precision, Coverage and F1 experimental values for iPACT

threshold	Precision	Coverage	F1
0.1	0.562	0.6294	0.59379352
0.2	0.572	0.6294	0.599328783
0.3	0.564	0.6441	0.601394587
0.4	0.550	0.6759	0.606485031
0.5	0.590	0.6040	0.596917923
0.6	0.620	0.5963	0.607919099
0.7	0.642	0.5320	0.581846678
0.8	0.670	0.5701	0.616026127
0.9	0.693	0.5503	0.61346079
1	0.730	0.5060	0.597702265

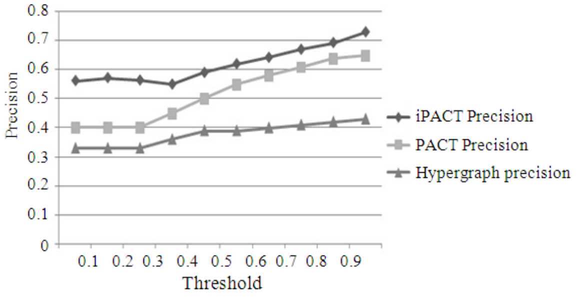


Fig. 4: Precision comparison

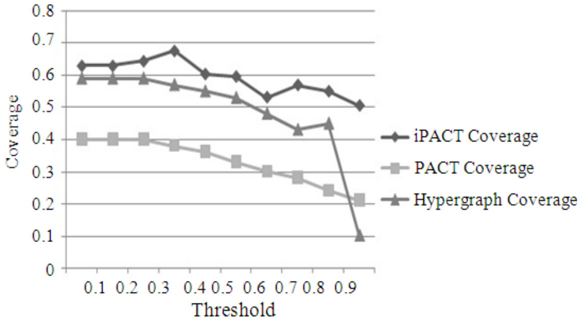


Fig. 5: Coverage comparison

For the evaluation purpose, iPACT is compared with two previous studies namely, PACT and Hypergraph on CTI dataset. Figures 4 and 5 depict the evaluation measurements for the prediction accuracy of iPACT against PACT and Hypergraph. Fig. 4 shows that the precision accuracy of iPACT is higher than PACT and Hypergraph. Higher precision means that iPACT recommendation engine produces accurate recommendations higher than the previous methods. Fig. 5 shows that the coverage accuracy of iPACT is better than PACT and Hypergraph. Better coverage indicates that the ability of the recommendation engine of iPACT to produce all of the pageviews that are likely to be visited by the user is better than the previous methods.

DISCUSSION

Neither of these measures individually is sufficient to evaluate the performance of the recommendation engine, however, they are both critical. This is particularly true in the context of e-commerce were recommendations are products. Low precision will likely result in unsatisfied users not interested in the recommended items. Low coverage will cause the site's inability to produce cross-sell relevant recommendations. The harmonic mean of both the precision and the coverage will produce an efficient measurement called F1. Fig. 6 relates the recommendation accuracy of iPACT compared to the findings of PACT and Hypergraph with sliding window equal to 3. With a recommendation threshold varies from 0.1 to 1.0, the F1 measurement as a performance evaluation shows that iPACT performs better and achieves higher prediction accuracy. This improvement in term of recommendation accuracy is because of the effectiveness of the prediction engine of the proposed iPACT recommendation system which ensures that the online component correctly classified the active sessions to the best web navigation profiles.

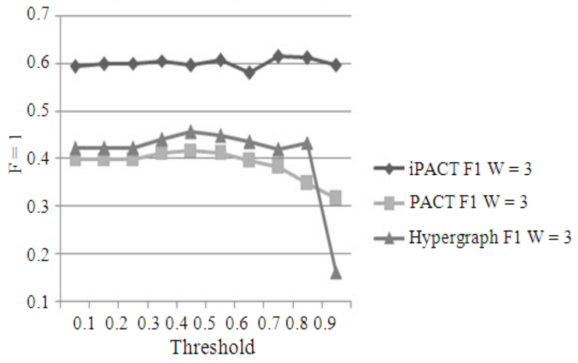


Fig. 6: F1 Comparison of recommendation accuracy

CONCLUSION

This study presented iPACT, an improved recommendation system based on PACT methodology. The improvement is due to the effectiveness of the prediction engine of the online phase of iPACT which shows better classification for the current user to the best web navigation profile represents his interests. Due to the navigation of many users and the change of their login time or interests, the web navigation profiles should be extracted again which is a time consuming. Incremental and adaptive navigation profiles will be more suitable for the prediction engine and is a key feature for the future works.

REFERENCES

AlMurtadha, Y.M., M.N.B. Sulaiman, N. mustapha and N.I. Udzir, 2010. Mining web navigation profiles for recommendation system. Inform. Technol. J., 9: 790-796. DOI: 10.3923/itj.2010.790.796

Barragáns-Martínez, A.B., E. Costa-Montenegro, J. Burguillo, M. Rey-López and F.A. Mikic-Fonte *et al*., 2010. A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition. Inform. Sci., 180: 4290-4311. DOI: 10.1016/j.ins.2010.07.024

Castellano, G., A.M. Fanelli and M.A. Torsello, 2011. NEWER: A system for NEuro-fuzzy web recommendation. Applied Soft Comput., 11: 793-806. DOI: 10.1016/j.asoc.2009.12.040

Dimopoulos, C., C. Makris, Y. Panagis, E. Theodoridis and A. Tsakalidis, 2010. A web page usage prediction scheme using sequence indexing and clustering techniques. Data Know. Eng., 69: 371-382. DOI: 10.1016/j.datak.2009.04.010

Forsati, R. and M.R. Meybodi, 2010. Effective page recommendation algorithms based on distributed learning automata and weighted association rules. Expert Syst. Appl., 37: 1316-1330. DOI: 10.1016/j.eswa.2009.06.010

Ganapathy, G. and P. Arunesh, 2010. Feature analysis of recommender techniques employed in the recommendation engines. J. Comput. Sci., 6: 748-755. DOI: 10.3844/jcssp.2010.748.755

Gao, M., K. Liu and Z. Wu, 2009. Personalisation in web computing and informatics: Theories, techniques, applications, and future research. Inform. Syst. Frontiers, 12: 607-629. DOI: 10.1007/s10796-009-9199-3

Göksedef, M. and S. Gündüz-Ögüdücü, 2010. Combination of web page recommender systems. Expert Syst. Appl., 37: 2911-2922. DOI: 10.1016/j.eswa.2009.09.046

Jalali, M., N. Mustapha, M.N. Sulaiman and A. Mamat, 2010. WebPUM: A Web-based recommendation system to predict user future movements. Expert Syst. Appl., 37: 6201-6212. DOI: 10.1016/j.eswa.2010.02.105

Kim, Y.S., B.J. Yum, J. Song and S.M. Kim, 2005. Development of a recommender system based on navigational and behavioral patterns of customers in e-commerce sites. Expert Syst. Appl., 28: 381-393. DOI: 10.1016/j.eswa.2004.10.017

Kumar, P.R. and A.K. Singh, 2010. Web structure mining: exploring hyperlinks and algorithms for information retrieval. Am. J. Applied Sci., 7: 840-845. DOI: 10.3844/ajassp.2010.840.845

Kusrini, S. Hartati, R. Wardoyo and A. Harjoko, 2010. Differential diagnosis knowledge building by using CUC-C4.5 framework. J. Comput. Sci., 6: 180-185. DOI: 10.3844/jcssp.2010.180.185

Park, Y.-J. and K.-N. Chang, 2009. Individual and group behavior-based customer profile model for personalized product recommendation. Expert Syst. Appl., 36: 1932-1939. DOI: 10.1016/j.eswa.2007.12.034

Ramadhan, H., M. Hatem, Z. Al-Khanjri and S. Kutti, 2005. A classification of techniques for web usage analysis. J. Comput. Sci., 1: 413-418. DOI: 10.3844/jcssp.2005.413.418

Ren, J.D.I. and X.L. Zhou, 2006. A new incremental updating algorithm for mining sequential patterns. J. Comput. Sci., 2: 318-321. DOI: 10.3844/jcssp.2006.318.321

Suhasini, P.S., K.S.R. Krishna and I.V.M. Krishna, 2008. Graph based segmentation in content based image retrieval. J. Comput. Sci., 4: 699-705. DOI: 10.3844/jcssp.2008.699.705

Sumathi, C.P., R.P. Valli and T. Santhanam, 2010. An application of session based clustering to analyze web pages of user interest from web log files. J. Comput. Sci., 6: 785-793. DOI: 10.3844/jcssp.2010.785.793

Vijayalakshmi, S. and V. Mohan, 2010. Mining sequential access pattern with low support from large pre-processed web logs. J. Comput. Sci., 6: 1293-1300. DOI: 10.3844/jcssp.2010.1293.1300