A Recommender System for Online Personalization in the WUM Applications

Mehrdad Jalali[1], Norwati Mustapha[2], Ali Mamat[2], Md. Nasir B Sulaiman[2]

***Abstract*—foreseeing of user future movements and intentions based on the users' clickstream data is a main challenging problem in Web based recommendation systems. Web usage mining based on the users' clickstream data has become the subject of exhaustive research, as its potential for web based personalized services, predicting user near future intentions, adaptive Web sites and customer profiling is recognized. A variety of the recommender systems for online personalization through web usage mining have been proposed. However, the quality of the recommendations in the current systems to predict users' future intentions systems cannot still satisfy users specially for long pattern of user activities in particular web sites. In this paper, to provide online predicting effectively, we develop a model for online predicting through web usage mining system and propose a novel approach for classifying user navigation patterns to predict users' future intentions. The approach is based on the using longest common subsequence (LCS) algorithm to classify current user activities to predict user next movement. We have tested our proposed model on the CTI datasets. The results indicate that the approach can improve the quality of the system for the predictions. Moreover, by using LCS, we can achieve more accurate recommendation for long patterns of the current user activities in the particular web sites.**

***Index Terms*—** Recommender Systems, Web Usage Mining, Longest Common Subsequence.

I. INTRODUCTION

Web based recommender systems use the opinions of a community of users to help individuals in that community more effectively identify content of interest from a potentially overwhelming set of choices [1]. Web-based recommender systems have been shown to greatly help Web users in navigating the Web, locating relevant and useful information, and receiving dynamic recommendations from Web sites on possible products or services that match their interests. Web usage mining is one of the main approaches used to build Web recommender systems. Meanwhile, the substantial increase in the number of websites presents a challenging task for webmasters to organize the contents of the websites to cater to the needs of users. Modeling and analyzing web navigation behavior is helpful in understanding what information of online users demand. Following that, the analyzed results can be seen as knowledge to be used in intelligent online applications, refining web site maps, web based personalization system and improving searching accuracy when seeking information. Nevertheless, an online navigation behavior grows each passing day, and thus extracting information intelligently from it is a difficult issue. Web usage mining refers to the automatic discovery and analysis of patterns in clickstream and associated data collected or generated as a result of user interactions with Web resources on one or more Web sites [2-4]. Web usage mining has been used effectively as an approach to automatic personalization and as a way to overcome deficiencies of traditional approaches such as collaborative filtering. The goal of personalization based on Web usage mining is to recommend a set of objects to the current (active) user, possibly consisting of links, ads, text, products, and so forth, tailored to the user's perceived preferences as determined by the matching usage patterns. This task is accomplished by matching the active user session with the usage patterns discovered through Web usage mining recommendation systems.

In this paper, to provide online predicting effectively, we develop a model for online predicting through web usage mining system and propose a novel approach for classifying user navigation patterns to predict users' future intentions. The approach is based on the using longest common (LCS) subsequence algorithm to classify current user activities to predict user next movement.

The rest of this paper is organized as follows: In section 2, we review recent research advances in web usage mining. Section 3 describes the system design and section 4 focuses on the user classifying based on the longest common subsequence. The system evaluation and the results of experimental evaluations are reported in sections 4 and 5. Finally, section 6 summarizes the paper and introduces future work.

II. RELATED WORK

Web based recommender systems are very helpful in directing the users to the target pages in particular web sites. Moreover, Web usage mining recommender systems have been proposed to predict user's intention and their navigation behaviors. In the following, we review some of the most significant WUM recommender systems and architecture that can be compared with our system.

Jalali et al. [5, 6] proposed a recommender system for navigation pattern mining through Web usage mining to predict user future movements. The approach is based on the graph partitioning clustering algorithm to model user navigation patterns for the navigation patterns mining phase. Furthermore, in the recommender phase, longest common

[1]Mehrdad Jalali is with department of software engineering of Islamic Azad University, Mashhad branch, Mashhad, Iran and he is a Ph.D. candidate of computer science in Universiti Putra Malaysia, Email: mehrdadjalali@ieee.org.

[2]Assistant Prof. Dr. Norwati Mustapha, Associate Prof. Dr. Md Nasir Sulaiman, and Associate Prof. Dr. Ali Mamat are with Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM, Selangor, Malaysia. Emails: {norwati,nasir,ali}@fsktm.upm.edu.my.

subsequence algorithm is utilized to classify current user activities to foresee user next movement.

Mobasher et al., present WebPersonalizer a system which provides dynamic recommendations, as a list of hypertext links, to users [7, 8]. The analysis is based on anonymous usage data combined with the structure formed by the hyperlinks of the site. Data mining techniques (i.e. clustering, association rules and sequential pattern discovery) are used in the preprocessing phase in order to obtain aggregate usage profiles. In this phase Web server logs are converted in clusters made up of sequences of visited pages, and cluster made up of set of pages with common usage characteristics. The online phase considers the active user session in order to find matches among the user's activities and the discovered usage profiles. Matching entries are then used to compute a set of recommendations which will be inserted into the last requested page as a list of hypertext links. WebPersonalizer is a good example of two-tier architecture for Personalization systems.

Baraglia and Palmerini proposed a WUM system called SUGGEST, that provide useful information to make easier the web user navigation and to optimize the web server performance [9, 10]. SUGGEST adopts a two levels architecture composed by an offline creation of historical knowledge and an online engine that understands user's behavior. As the requests arrive at this system module it incrementally updates a graph representation of the Web site based on the active user sessions and classifies the active session using a graph partitioning algorithm. Potential limitation of this architecture might be: a) the memory required to store Web server pages is quadratic in the number of pages. This might be a severe limitation in large sites made up of millions of pages; b) it does not permit us to manage Web sites made up of pages dynamically generated.

All of these works attempt to find architecture and algorithm to improve quality of the personalized recommendation, but the recommendations still do not meet satisfaction. In our work, we advance a model and propose novel approach to predict user intention in the near future request.

III. SYSTEM DESIGN

In this paper, a model is proposed to predict user's future requests and intentions. Generally, Web Usage Mining recommender systems consist of two main phase. Mining of user navigation patterns has done in offline phase of the recommender system. In the online phase, the recommender system predicts a set of web pages as user next intentions. We have done navigation patterns mining based on the graph partitioning algorithm as offline phase of a recommender system [11] . Figure 1 illustrates the model of the system.

According to different part of the proposed model, online components in a majority of prediction engine are done in the online phase of the system to predict user's future requests.

Classifying of user's current activities based on navigation patterns in the particular web site is the main objective of the prediction engine. In addition, creating a list of recommendations web pages as prediction is another objective in online phase. In the following, we state the online components.

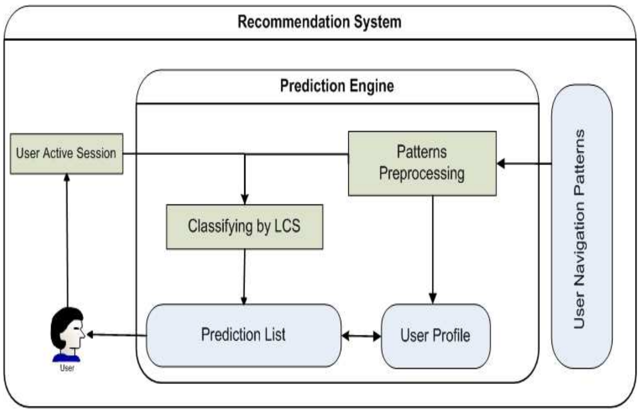

Figure 1: Model of the system

As the first objective, the prediction engine is used to classify user navigation patterns and predicts users' future requests. For this purpose, we propose a novel approach to classify current user activity. In order to classify user's active session, we look for the navigation pattern that includes the larger number of similar web pages in the session. Pattern search approaches can be utilized to find similar web pages between the current active session and navigation patterns. The longest common subsequence (LCS) algorithm is to find the longest subsequence common to all sequences in a set of sequences (often just two). In the next section, we describe about the longest common Subsequence algorithm.

The second objective of this component is computing a recommendation set for the current session, consisting of links to pages that the user may want to visit based on similar usage patterns. The recommendation set essentially represents a "short-term" view of potentially useful links based on the user's navigational activity through the site. These recommended links are then added to the last page in the session accessed by the user before that page is sent to the user browser.

IV. USER CLASSIFYING THROUGH LONGEST COMMON SUBSEQUENCE

The problem of comparing two sequences $\vec{\alpha}$ and $\vec{\beta}$ to determine their similarity is one of the fundamental problems in pattern matching. One of the basic forms of the problem is to determine the longest common subsequence (LCS) of $\vec{\alpha}$ and $\vec{\beta}$. The LCS string comparison metric measures the subsequence of maximal length common to both sequences [12].

Formally, given a sequence $\vec{\alpha} = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$, another sequence $\vec{\gamma} = \langle \Upsilon_1, \Upsilon_{2,\ldots,} \Upsilon_n \rangle$ is a subsequence of $\vec{\alpha}$ if there exists a strictly increasing sequence $\langle j_1, j_{2,\ldots,} j_n \rangle$ of indices of $\vec{\alpha}$ such that for all i=1,2,..,l, we have $\alpha_{ji} = \Upsilon_i$. Given two sequence $\vec{\alpha}$ and $\vec{\beta}$, we say that $\vec{\gamma}$ is common subsequence of $\vec{\alpha}$ and $\vec{\beta}$ if $\vec{\gamma}$ is a subsequence of both $\vec{\alpha}$ and $\vec{\beta}$ .We are interested in finding the maximum-length or longest common subsequence (LCS) given two paths or sequence of page-visits $\vec{\alpha} = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$, $\vec{\beta} = \langle \beta_1, \beta_2, \ldots, \beta_m \rangle$.

The LCS has a well-studied optimal sub-structure property

as given by the following:
Theorem 1: Let $\vec{\alpha} = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$ and $\vec{\beta} = \langle \beta_1, \beta_2, \ldots, \beta_m \rangle$ be sequences, and let $\vec{\gamma} = \langle \Upsilon_1, \Upsilon_2, \ldots, \Upsilon_n \rangle$ be any LCS of $\vec{\alpha}$ and $\vec{\beta}$.
If $\alpha_n = \beta_m$, then $\Upsilon_l = \alpha_n = \beta_m$ and $\vec{\gamma}_{l-1}$ is a LCS of $\vec{\alpha}_{n-1}$ and $\vec{\beta}_{m-1}$.
If $\alpha_n \neq \beta_m$, then $\Upsilon_l \neq \alpha_n$ implies $\vec{\gamma}$ is a LCS of $\vec{\alpha}_{n-1}$ and $\vec{\beta}$.
If $\alpha_n \neq \beta_m$, then $\Upsilon_l \neq \beta_m$ implies $\vec{\gamma}$ is a LCS of $\vec{\alpha}$ and $\vec{\beta}_{m-1}$.
where $\vec{\alpha}_{n-1} = \langle \alpha_1, \alpha_2, \ldots, \alpha_{n-1} \rangle$, $\vec{\beta}_{m-1} = \langle \beta_1, \beta_2, \ldots, \beta_{m-1} \rangle$ and $\vec{\gamma}_{l-1} = \langle \Upsilon_1, \Upsilon_2, \ldots, \Upsilon_{n-1} \rangle$.

The proof of this theorem can be found in [13]. Efficient recursive algorithms to compute the LCS exist using this property of the LCS [14]. We shall not go into the details of the algorithms since a lot of literature already exists on the topic[14-16].

Definition 1: Let s1 and *s*2 be two sequences. |*LCS (s*1, *s*2*)*| is the size of the longest common subsequence between *s*1 and *s*2. The degree of similarity between *s*1 and *s*2 is defined as below:

$$Sim_{LCS=} \frac{2 \times |LCS(S1,S2)|}{|S1| + |S2|} \qquad (1)$$

Pattern search algorithm can be utilized to find navigation patterns based on the current user activities to predict and recommend user future's request. We apply a pattern search algorithm namely Longest Common Subsequences (LCS) in the recommendation part of the system. In this paper, there are several steps to create recommendations set based on the current user's session in the online phase of the system which will be explained below.

A. Data Pretreatment for recommendation

Preprocessing for both current active session and navigation patterns is done in the first step of the prediction engine. In this step the prediction engine performs is preparing data for applying LCS algorithm with take into account the efficiency of the algorithm.

In this study, the current active session *S* is represented as a vector:

$$\vec{S} = \left\langle P1, P2, \ldots, Pm \right\rangle \qquad (2)$$

where Pi=n, and n is a unique numeric value that we assigned to each web pages in the offline phase. If user visits a web page, the system replaces it with a predefined unique numeric value.

Fix-size sliding window is utilized over the current active session to capture the current user's activities. We call this sliding window, the user's active session window. We consider the mean of web pages in each session of dataset as user's active session window.

Furthermore, in the proposed algorithm, web pages inside the user's active session window have to put in order according to the numeric values.

Here, we present each cluster that created in the offline phase as set of navigation patterns.

$$n\vec{p} = \left\langle n\vec{p}1, n\vec{p}2, \ldots, n\vec{p}m \right\rangle \qquad (3)$$

where $n\vec{p}i$, is a set of *k* web pages as a navigation pattern and show as below :

$$n\vec{p}i = \left\langle P1, P2, \ldots, Pk \right\rangle \qquad (4)$$

where $1 \le i \le n$, and *pi* is a web pages in a navigation pattern. Moreover, $n\vec{p}i$ have to put in order as same as user's active session window.

B. User classifying based on LCS algorithm

There are two sets, navigation patterns $n\vec{p}i$ and active session window $\vec{S}$ as input of this step. Classifying algorithm attempts to find a navigation pattern (Cluster) by utilizing longest common subsequences algorithm. A navigation pattern with the highest degree of similarity is found according to the LCS algorithm to predict next user's activities and create a recommendation set.

C. *Predict user next's intention*

In this step of the algorithm, a set of web pages shows to the user as recommendation set. A recommendation engine attempts to show a set of web pages to the current user after the system finds a navigation pattern with the highest degree of similarity. Meanwhile, web pages in the recommendation set are ranked in terms of degree of connectivity between web pages in adjacency matrix *M* that created in the offline phase. Moreover, for improving the quality of recommendation, the recommendation engine shows only the web pages by highest degree of connectivity and the rest of web pages were not be considered in the recommendation set. The new recommendation set is created by next user's movement in the web site. In this case, after each user's activity, new user session window is created.

V. System Evaluation

A variety of techniques are used to measure the performance of the recommender systems. Some experimentations have been done for characterize the quality of the recommendation. In order to evaluate effectiveness of the proposed system several tests should be conducted for both online phase and offline phase.

Measuring the accuracy of the predictions and predicting in the recommender system needs to characterize the quality of the results obtained. As we described evaluation method in the offline phase, to measure the quality of recommendation, we use second half of the dataset after the dataset divided into two halves; training set and evaluation set. Each navigational pattern *npi* (a session in the dataset) in the evaluation set is divided into two parts. The first *n* pageviews in *npi* are used for generating predictions, whereas, the remaining part of *npi* is used to evaluate the generated predictions. The active session window is the part of the user's navigational patterns used by the prediction engine in order to produce a prediction set. We call this part of the navigational pattern *np* the active session with respect to np, denoted by as_{np}. The prediction engine takes as_{np} and a recommendation threshold τ as inputs and produces a set of pageviews as a prediction list. Recommendation threshold τ is the *MinFreq*. We denote this prediction set by $P(as_{np}, \tau)$. The set of pageviews $P(as_{np}, \tau)$ can now be compared with the remaining $|np| - n$, pageviews in np. We denote this part of np by $eval_{np}$. Our comparison of these sets is based on 3 different metrics, namely, *accuracy*, *coverage* and *F1 measure*. The *accuracy* of prediction set is defined as:

$$Accuracy(P(as_{np},\tau)) = \frac{|P(as_{np},\tau) \cap eval_{np}|}{P(as_{np},\tau)} \tag{5}$$

where, $|P(as_{np},\tau) \cap eval_{np}|$ is number of web pages that these are common in the prediction list and evaluation set. *Accuracy* is Number of relevant web pages retrieved divide by the total number of web pages in recommendations set.

In the other hand, accuracy measures the degree to which the prediction engine produces accurate recommendations.

Another evaluation parameter in the online phase is *coverage* that is defined as:

$$Coverage(P(as_{np},\tau)) = \frac{|P(as_{np},\tau) \cap eval_{np}|}{eval_{np}} \tag{6}$$

Coverage is number of relevant web pages retrieved divide by the total number of web pages that actually belong to the user sessions. In the other hand, coverage measures the ability of the prediction engine to produce all of the pageviews that are likely to be visited by the user.

The *F1 measure* attains its maximum value when both accuracy and coverage are maximized. Finally, for a given prediction threshold τ, the mean over all navigational pattern in the evaluation set is computed as the overall evaluation score for each measure.

$$F1 = \frac{2 \times Accuracy(P(as_{np},\tau)) \times Coverage(P(as_{np},\tau))}{Accuracy(P(as_{np},\tau)) + Coverage(P(as_{np},\tau))} \tag{7}$$

All parameters attempt to measure the quality of recommendation in the range of *MinFreq* between 0 and 1.

VI. Experimental Evaluation

In order to evaluate the performance of the proposed system, experimentation has been conducted for prediction of the user's next request by classifying algorithm based on longest common subsequence.

All evaluation tests have run on a dual processor Intel® Core™ Duo CPU 2.4 GHz with 3.23 GBytes of RAM, operating system Windows XP. Our implementations have run on .Net framework 2 and VB.net and C#.net have been used for coding the proposed system.

For our experiments, it is necessary to use such a dataset that allows us to evaluate the recommendations. Our experiments have been conducted on DePaul University CTI logs file dataset (www.cs.depaul.edu).

All evaluations for the creating user navigation patterns have been done based on the CTI dataset in the our previous work [11].

The length of the active session window is important to classify current active session in the proposed recommender system. The average number of web pages in a session can be used for considering the length of active session. As shown in Figure 2 the percentage of the sessions formed by a predefined number of pages quickly decreases when the minimum number of pages in a session increases. Moreover, for CTI dataset, average length of a user session is about 3 pages. Therefore, we still have almost half of all the sessions, we choose this value as the minimum length for an active session to be classified.

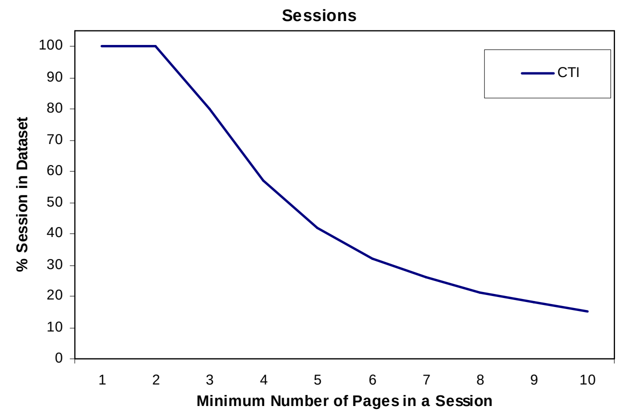

Figure 2: Minimum Number of pages in a session

In this section, we measure the quality of the recommendation generated by prediction engine during the online phase. Sessions found in one half of the both datasets are submitted to the prediction engine to classify current user's activities and to generate recommendation. Subsequently, the overlapping between the generated prediction $P(as_{np}, \tau)$ and the session pages $eval_{np}$ is computed by using expression (5) that introduce in section 5. Finally, the percentage of all predictions figures out as quality of the predicting in the proposed recommender system.

In this study, we evaluate the quality of the recommendations between the navigation pattern created by proposed method [11] and classifying method based on LCS algorithm by the previous algorithms [11]. Three parameters have been measured to verify the quality of the predictions; accuracy, coverage and F1.

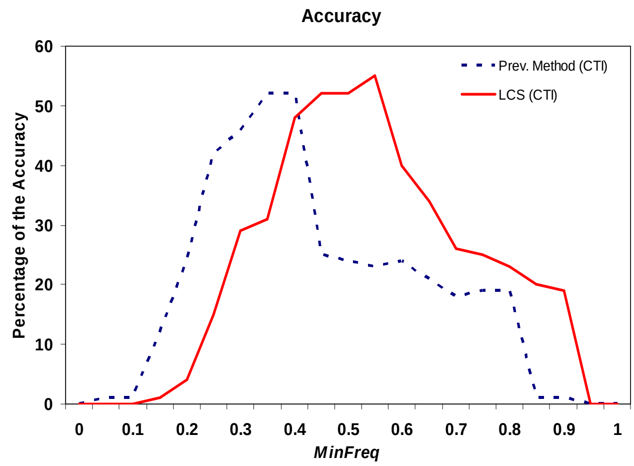

Figure 3: Accuracy of the recommendations

Figure 3 depicts the accuracy of the proposed system for

MinFreq ranging from 0 to 1 for the CTI dataset. Minimum frequency is a parameter for filtering the weights which are below a constant value, named as *MinFreq*. The edges of the graph whose values are less than *MinFreq* are inadequately correlated and thus they are not considered by the DFS graph search algorithm. This parameter is used in the navigation pattern mining as threshold.

The percentage of the accuracy for the proposed system achieves the best results when we choose the value of the *MinFreq* to be around 0.55 for CTI dataset. The results illustrate the approach can improve the accuracy of the recommendation in a great deal of difference values of the *MinFreq* against the previous work.

Figure 4 depicts the coverage of the proposed system for *MinFreq* ranging from 0 to 1. The coverage achieves high percentage for the lower values of the *MinFreq*. In this case, for the lower values of the *MinFreq* by classifying current user's activities, the prediction engine can find the clusters with large size that many web pages inside the cluster belong to the actual user's activities. Moreover, similar reasoning can be used for the higher values of the *MinFreq* that decrease the percentage of the coverage. The results illustrate the approach can improve coverage of the recommender system in comparison with previous work.

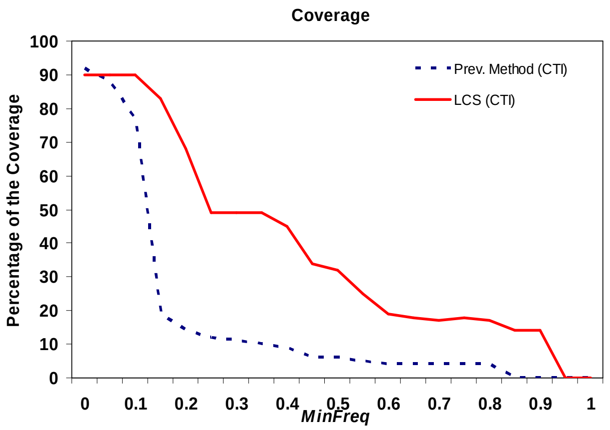

Figure 4: Coverage of the recommendations

Figure 5 depict the F1 measure for *MinFreq* ranging from 0 to 1 for the CTI dataset. In the most time, F1 achieves higher percentages for the proposed system than the previous work.

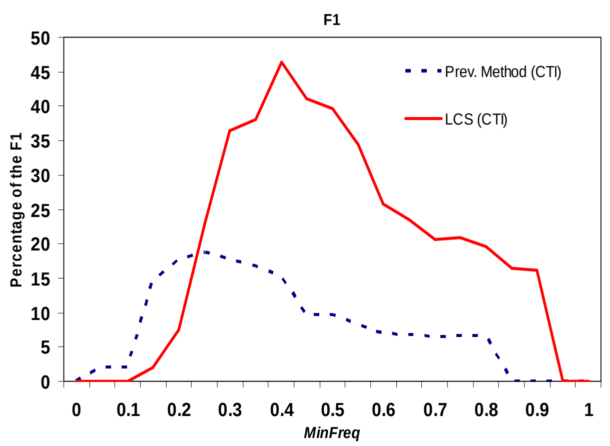

Figure 5: F1 measure of the recommendations

For all experiments, we run paired t-test with 95% confidence .Paired t-test is used to compare means on the same or related subject over time or in the differing circumstances. On the other hand, the t-test assesses whether the means of two groups are statistically different from each other. This analysis is appropriate whenever you want to compare the means of two groups. The assumption for using the paired t-test is that the observed data are from the same subject or from a matched subject and are drawn from a population with a normal distribution. In this study, a paired t-test is carried out to compare the experimental results for the F1 measure. The mean of F1 measure for the CTI dataset is 19.5 for the proposed approach and 7.9 for the previous work. The two-tail *p*-value for the paired t-test is 0.0006 that achieves significant value (*p*-value<=0.05).

In this paper, we also test the scalability based on the current active session (window size). We start the experiment with window size 2, and according to this size, we measure the accuracy of the WebPUM system. Moreover, in the each trail we add one page to current user active session (window). On the other hand, window size increases in each trail. As we shown in Figure 6, the percentage of the accuracy increases, if the window size changes between 2 and 13. In the most time, the accuracy is stable in the previous work according to the difference window size. However, the weakness of increasing the window size is that the number of active session which can be considering for the recommendation system is decrease and the system cannot prepare prediction for a great deal of active sessions.

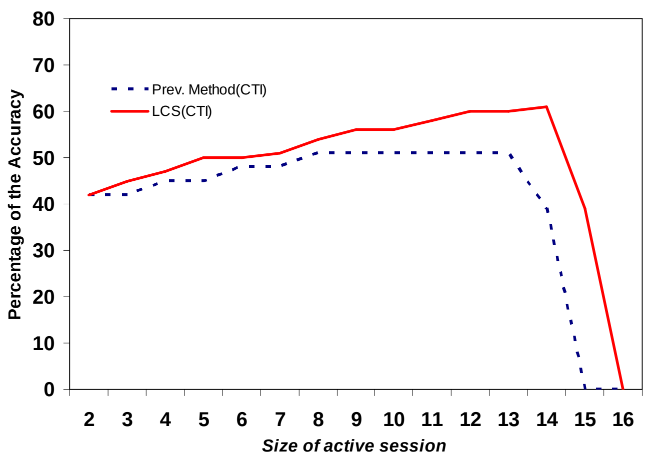

Figure 6: The accuracy based on size of active session (window size)

Based on experimental results, it indicates that our approach to classify current user's activities to predict user's next request can improve the quality of the predictions in the web usage mining recommender systems.

VII. Conclusions and Future Work

In this paper, we advanced a system and proposed a novel approach to classify the user navigation pattern for online predicting users' future intentions through mining of web server logs. To classify current user activities, we used longest common subsequences algorithm to predict user near future movement. We used some evaluation methodologies that can evaluate the quality of the clusters found and quality of the recommendations. The experimental results show that our approach can improve the quality of clustering for user the navigation pattern and quality of recommendation for the CTI dataset. Moreover, by using LCS, we achieved more accurate recommendation for long patterns of the current user activities in the particular web sites.

There are some aspects in that can be improved in our system. For instance, we can take into account the semantic knowledge about underlying domain to improve the quality of the recommendation. On the other hand, integrating semantic web and web usage mining can achieve best recommendations in the dynamic huge web sites. Moreover, we would perfect the system and let it serves for actual users to the best of its abilities.

REFERENCES

[1] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM,* vol. 40, pp. 56-58, 1997.

[2] R. Cooley, B. Mobasher, and J. Srivastava, "Web mining: information and pattern discovery on the World Wide Web," 1997, pp. 558-567.

[3] J. Wang and I. NetLibrary, *Encyclopedia of Data Warehousing and Mining*: Idea group Reference, 2006.

[4] J. Srivastava, R. Cooley, M. Deshpande, and P. N. Tan, "Web usage mining: discovery and applications of usage patterns from Web data," *ACM SIGKDD Explorations Newsletter,* vol. 1, pp. 12-23, 2000.

[5] M. Jalali, N. Mustapha, M. N. B. Sulaiman, and A. Mamat, "OPWUMP: An Architecture for Online Predicting in WUM-Based Personalization System," *Communications in Computer and Information Science,Advances in Computer Science and Engineering,Springer Berlin Heidelberg,* vol. 6, pp. 838–841, 2008.

[6] M. Jalali, N. Mustapha, N. B. Sulaiman, and A. Mamat, "A Web Usage Mining Approach Based on LCS Algorithm in Online Predicting Recommendation Systems," in *12th International on Information Visualisation,IV'08*, London, UK, 2008, pp. 302-307.

[7] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic personalization based on Web usage mining," *Communications of the ACM,* vol. 43, pp. 142-151, 2000.

[8] M. Nakagawa and B. Mobasher, "A hybrid web personalization model based on site connectivity," 2003, pp. 59–70.

[9] R. Baraglia and F. Silvestri, "Dynamic personalization of web sites without user intervention," *Communications of the ACM,* vol. 50, pp. 63-67, 2007.

[10] R. Baraglia and F. Silvestri, "An Online Recommender System for Large Web Sites," 2004, pp. 199-205.

[11] M. Jalali, N. Mustapha, M. N. Sulaiman, and A. Mamat, "Web User Navigation Pattern mining approach based on Graph Partitioning algorithm " *Journal of Theoretical and Applied Information Technology* vol. 4, pp. 1125-1130, 2008.

[12] A. Apostolico, "String editing and longest common subsequences," *Handbook of Formal Languages,* vol. 2, pp. 361–398, 1997.

[13] T. H. Gormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms," *MIT Press,* vol. 44, pp. 97,138, 1990.

[14] V. Dancik, *Expected Length of Longest Common Subsequences*: University of Warwick, 1994.

[15] A. V. Aho, D. S. Hirschberg, and J. D. Ullman, "Bounds on the complexity of the longest common subsequence problem," 1974, pp. 104-109.

[16] D. S. Hirschberg, "Algorithms for the Longest Common Subsequence Problem," *Journal of the ACM (JACM),* vol. 24, pp. 664-675, 1977.