Integrating Recommendation Models for Improved Web Page Prediction Accuracy

Faten Khalil[1] **Jiuyong Li**[2] **Hua Wang**[1]

[1] Department of Mathematics & Computing
University of Southern Queensland,
Toowoomba, Australia, 4350,
Email: {khalil and wang}@usq.edu.au

[2] School of Computer & Information Science
University of South Australia,
Mason Lakes, Australia,
Email: Jiuyong.Li@unisa.edu.au

Abstract

Recent research initiatives have addressed the need for improved performance of Web page prediction accuracy that would profit many applications, e-business in particular. Different Web usage mining frameworks have been implemented for this purpose specifically Association rules, clustering, and Markov model. Each of these frameworks has its own strengths and weaknesses and it has been proved that using each of these frameworks individually does not provide a suitable solution that answers today's Web page prediction needs. This paper endeavors to provide an improved Web page prediction accuracy by using a novel approach that involves integrating clustering, association rules and Markov models according to some constraints. Experimental results prove that this integration provides better prediction accuracy than using each technique individually.

Keywords: Web page prediction, association rules, clustering, Markov model.

1 Introduction

Web page access prediction gained its importance from the ever increasing number of e-commerce Web information systems and e-businesses. Web page prediction that involves personalizing the Web users' browsing experiences assists Web masters in the improvement of the Web site structure, and helps Web users in navigating the site and accessing the information they need. Various attempts have been exploited to achieve Web page access prediction by preprocessing Web server log files and analyzing Web users' navigational patterns. The most widely used approach for this purpose is Web usage mining that entails many techniques like Markov model, association rules and clustering (Srivastava et al. 2000).

- Markov models are the most effective techniques for Web page access prediction and many researchers stress the importance in the field (Bouras & Konidaris 2004, chen et al. 2002, Deshpande & Karypis 2004, Eirinaki et al. 2005, Zhu et al. 2002). Other researchers use Markov models to improve the Web server access efficiency either by using object prefetching (Pons 2006) or by helping reduce the Web server overhead (Mathur & Apte 2007). Lower order Markov models are known for their low accuracy due to the limited availability of users' browsing history. Higher order Markov models achieve higher accuracy but are associated with higher state space complexity.
- Association rule mining is a major pattern discovery technique (Mobasher et al. 2001). The original goal of association rule mining is to solve market basket problem but the applications of association rules are far beyond that. Using association rules for Web page access prediction involves dealing with too many rules and it is not easy to find a suitable subset of rules to make accurate and reliable predictions (Kim et al. 2004, Mobasher et al. 2001, Yong et al. 2005).
- Although clustering techniques have been used for personalization purposes by discovering Web site structure and extracting useful patterns (Adami et al. 2003, Cadez et al. 2003, Papadakis & Skoutas 2005, Rigou et al. 2006, Strehl et al. 2000), usually, they are not very successful in attaining good results. Proper clustering groups users sessions with similar browsing history together, and this facilitates classification. However, prediction is performed on the cluster sets rather than the actual sessions.

Therefore, there arises a need for improvement when using any of the aforementioned techniques. This paper integrates all three frameworks together, clustering, association rules and Markov model, to achieve better Web page access prediction performance specifically when it comes to accuracy.

Web page access prediction can be useful in many applications. The improvement in accuracy can make a change in the Web advertisement area where a substantial amount of money is paid for placing the correct advertisements on Web sites. Using Web page access prediction, the right ad will be predicted according to the users' browsing patterns. Also, using the Web users' browsing patterns Web page access prediction helps Web administrators restructure the Web sites to improve site topology and user personalization as well as market segmentation. Web page access prediction is also helpful for caching the predicted page for faster access, for improved Web page ranking and for improving browsing and navigation orders.

2 Related Work

A number of researchers attempted to improve the Web page access prediction precision or coverage by combining different recommendation frameworks. For instance, many papers combined clustering with association rules (Lai & Yang 2000, Liu et al. 2001). Lai & Yang (2000) have introduced a customized marketing on the Web approach using a combination of clustering and association rules. The authors collected information about customers using forms, Web server log files and cookies. They categorized customers according to the information collected. Since k-means clustering algorithm works only with numerical data, the authors used PAM (Partitioning Around Medoids) algorithm to cluster data using categorical scales. They then performed association rules techniques on each cluster. They proved through experimentations that implementing association rules on clusters achieves better results than on non-clustered data for customizing the customers' marketing preferences. Liu et al. (2001) have introduced MARC (Mining Association Rules using Clustering) that helps reduce the I/O overhead associated with large databases by making only one pass over the database when learning association rules. The authors group similar transactions together and they mine association rules on the summaries of clusters instead of the whole data set. Although the authors prove through experimentation that MARC can learn association rules more efficiently, their algorithm does not improve on the accuracy of the association rules learned.

Other papers combined clustering with Markov model (Cadez et al. 2003, Zhu et al. 2002, Lu et al. 2005). Cadez et al. (2003) partitioned site users using a model-based clustering approach where they implemented first order Markov model using the Expectation-Maximization algorithm. After partitioning the users into clusters, they displayed the paths for users within each cluster. They also developed a visualization tool called WebCANVAS based on their model. Zhu et al. (2002) construct Markov models from log files and use co-citation and coupling similarities for measuring the conceptual relationships between Web pages. CitationCluster algorithm is then proposed to cluster conceptually related pages. A hierarchy of the Web site is constructed from the clustering results. The authors then combine Markov model based link prediction to the conceptual hierarchy into a prototype called ONE to assist users' navigation. Lu et al. (2005) were able to generate Significant Usage Patterns (SUP) from clusters of abstracted Web sessions. Clustering was applied based on a two-phase abstraction technique. First, session similarity is computed using Needleman-Wunsch alignment algorithm and sessions are clustered according to their similarities. Second, a concept-based abstraction approach is used for further abstraction and a first order Markov model is built for each cluster of sessions. SUPs are the paths that are generated from first order Markov model with each cluster of user sessions.

Combining association rules with Markov model is novel to our knowledge and only few of past researches combined all three models together (Kim et al. 2004). Kim et al. (2004) improve the performance of Markov model, sequential association rules, association rules and clustering by combining all these models together. For instance, Markov model is used first. If MM cannot cover an active session or a state, sequential association rules are used. If sequential association rules cannot cover the state, association rules are used. If association rules cannot cover the state, clustering algorithm is applied. Kim et al. (2004) work improved recall and it did not improve the Web page prediction accuracy. Our work proves to outperform previous works in terms of Web page prediction accuracy using a combination of clustering, association rules and Markov model techniques.

3 Related Methods

3.1 Clustering

This paper introduces a new model called Integrated Prediction Model, or IPM, that integrates clustering, Markov model and association rules mining frameworks in order to improve the Web page access prediction accuracy. The first problem encountered in this paper is the grouping of such sessions into k number of clusters in order to improve the Markov model prediction accuracy. Performing clustering tasks can be tedious and complex due to the increased number of clustering methods and algorithms. Clustering could be hierarchical or non-hierarchical (Jain et al. 1999), distance-based or model-based (Zhong & Ghosh 2003), and supervised or unsupervised (Eick et al. 2004). For the purpose of this paper, we use a straightforward implementation of the k-means clustering algorithm which is distance-based, based on user sessions, unsupervised and partitional non-hierarchical. K-means clustering algorithm involves the following:

1. defining a set of sessions (n-by-p data matrix) to be clustered where n represents sessions and p represents pages,
2. defining a chosen number of clusters (k) and
3. randomly assign a number of sessions to each cluster.

K-means clustering then repeatedly calculates the mean vector for all items in each cluster and reassigns the items to the cluster whose center is closest to the session until there is no change for all cluster centers. Because the first clusters are created randomly, k-means runs different times each time it starts from a different point giving different results. The different clustering solutions are compared using the sum of distances within clusters. In this paper, clusters were achieved using MatLab that considers the clustering solution with the least sum of distances. k-means clustering depends greatly on the number of clusters (k), the number of runs and the distance measure used. There exists a variety of distance measures, in particular, Euclidean, Squared Euclidean, City Block, Hamming, Cosine and Correlation (Strehl et al. 2000). In this paper we use Cosine distance measure that yields better clustering results than the other distance measures and is a direct application of the extended Jaccard coefficient (Strehl et al. 2000, Halkidi et al. 2003, Casale 2005).

3.2 Markov Model

After dividing user sessions into a number of clusters using cosine distance measure, Markov model analysis are carried out on each of the clusters. Markov models are used in the identification of the next page to be accessed by the Web site user based on the sequence of previously accessed pages (Deshpande & Karypis 2004). Let $P = \{p1, p2, \ldots, pm\}$ be a set of pages in a Web site. Let W be a user session including a sequence of pages visited by the user in a visit. Assuming that the user has visited l pages, then prob$(p_i|W)$ is the probability that the user visits pages p_i next. Page p_{l+1} the user will visit next is estimated by:

$$P_{l+1} = \text{argmax}_{p \in \mathbb{P}}\{P(P_{l+1} = p|W)\}$$
$$= \text{argmax}_{p \in \mathbb{P}}\{P(P_{l+1} = p|p_l, p_{l-1}, \ldots, p_1)\} \quad (1)$$

This probability, $prob(p_i|W)$, is estimated by using all sequences of all users in history (or training data), denoted by W. Naturally, the longer l and the larger W, the more accurate $prob(p_i|W)$. However, it is infeasible to have very long l and large W and it leads to unnecessary complexity. Therefore, a more feasible probability is estimated by assuming that the sequence of the Web pages visited by users follows a Markov process that imposes a limit on the number of previously accessed pages k. In other words, the probability of visiting a page p_i does not depend on all the pages in the Web session, but only on a small set of k preceding pages, where $k << l$.

The equation becomes:

$$P_{l+1} = \text{argmax}_{p \in \mathbb{P}}\{P(P_{l+1} = p|p_l, p_{l-1}, \ldots, p_{l-(k-1)})\} \quad (2)$$

where k denotes the number of the preceding pages and it identifies the order of the Markov model. The resulting model of this equation is called the all k^{th} order Markov model. Of course, the Markov model starts calculating the highest probability of the last page visited because during a Web session, the user can only link the page he is currently visiting to the next one. The probability of $P\left(p_i|S_j^k\right)$ is estimated as follows from a history (training) data set.

$$P\left(p_i|S_j^k\right) = \frac{\text{Frequency}\left(\left\langle S_j^k, p_i\right\rangle\right)}{\text{Frequency}\left(S_j^k\right)} \quad . \quad (3)$$

This formula calculates the conditional probability as the ratio of the frequency of the sequence occurring in the training set to the frequency of the page occurring directly after the sequence.

The fundamental assumption of predictions based on Markov models is that the next state is dependent on the previous k states. The longer the k is, the more accurate the predictions are. However, longer k causes the following two problems: The coverage of model is limited and leaves many states uncovered; and the complexity of the model becomes unmanageable (Deshpande & Karypis 2004). Therefore, the following are three modified Markov models for predicting Web page access.

1. All k^{th} Markov model: This model is to tackle the problem of low coverage of a high order Markov model. For each test instance, the highest order Markov model that covers the instance is used to predict the instance. For example, if we build an all 4-Markov model including 1-, 2-, 3-, and 4-, for a test instance, we try to use 4-Markov model to make prediction. If the 4-Markov model does not contain the corresponding states, we then use the 3-Markov model, and so forth (Pitkow & Pirolli 1999).

2. Frequency pruned Markov model: Though all k^{th} order Markov models result in low coverage, they exacerbate the problem of complexity since the states of all Markov models are added up. Note that many states have low statistically predictive reliability since their occurrence frequencies are very low. The removal of these low frequency states affects the accuracy of a Markov model. However, the number of states of the pruned Markov model will be significantly reduced.

3. Accuracy pruned Markov model: Frequency pruned Markov model does not capture factors that affect the accuracy of states. A high frequent state may not present accurate prediction. When we use a means to estimate the predictive accuracy of states, states with low predictive accuracy can be eliminated. One way to estimate the predictive accuracy using conditional probability is called confidence pruning. Another way to estimate the predictive accuracy is to count (estimated) errors involved, called error pruning.

In this paper, we employ the frequency pruned Markov model. When choosing the Markov model order, our aim is to determine a Markov model order that leads to high accuracy with low state space complexity. Figure 1 reveals the increase of precision as the frequency pruned Markov model increases using the four data sets introduced in section 5 below. On the other hand, table 1 and table 2 show the increase of the state space complexity as the order of all k^{th} and frequency pruned Markov model increases for all four data sets. The frequency pruned Markov model orders, and as it has been proposed by (Deshpande & Karypis 2004), does not increase the prediction accuracy significantly. It rather plays a major role in decreasing the state space complexity. Based on this information, we use the 2-FP order Markov model because it has better accuracy than that of the 1-FP order Markov model without the drawback of the state space complexity associated with higher order Markov models.

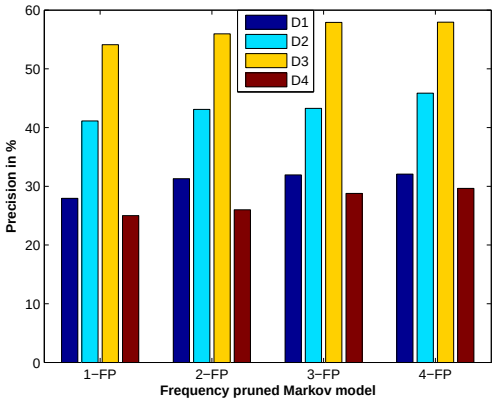

Figure 1: Precision of all 1-, 2-, 3- and 4- frequency pruned Markov model orders.

Table 1: Number of states of all 1- to 4- Markov model orders.

	1-MM	2-MM	3-MM	4-MM
D1	1945	39162	72524	101365
D2	1036	25060	89815	128516
D3	674	21392	50971	83867
D4	2054	34469	90123	131106

Table 2: Number of states of frequency pruned Markov model orders.

	1-FP	2-FP	3-FP	4-FP
D1	745	9162	14977	17034
D2	502	6032	18121	22954
D3	623	5290	11218	13697
D4	807	7961	19032	23541

3.3 Association Rules

The final step in the training process is to generate global association rules from the original data. Association rules are mainly defined by two metrics: support and confidence. Let A be a subsequence of W, and p_i be a page. We say that W supports A if A is a subsequence of W, and W supports $\langle A, p_i \rangle$ if $\langle A, p_i \rangle$ is a subsequence of W. The support for sequence A is the fraction of sessions supporting A in the data set D as follows:

$$\sigma = supp(A) = \frac{\{W \in D : A \subseteq W\}}{D} \tag{4}$$

The confidence of the implication is:

$$\alpha = conf(A) = \frac{supp(\langle A, P \rangle)}{supp(A)} \tag{5}$$

An implication is called an association rule if its support and confidence are not less than some user specified minimum thresholds. The selection of parameter values for σ and α usually has to be based on experience or even resorts to try and error. The most common association mining algorithm is Apriori algorithm (Agrawal & Srikant 1994). The main problem of mining association rules is composed of two steps:

1. Discovery of large itemsets.
2. Using the large itemsets to generate the association rules.

The second step is simple and the overall performance of mining association rules is determined by the first step. Apriori (Agrawal & Srikant 1994) addresses the issue of discovering large itemsets. In each iteration, Apriori constructs a candidate set of large itemsets, counts the number of occurrences of each candidate and determines the large itemsets based on a predetermined minimum support and confidence thresholds. In the first iteration, Apriori scans all the transactions to count the number of occurrences for each item and based on the minimum support threshold (σ), the first large itemset is determined. Therefore, the cost of the first iteration is $O(D)$. Next, the second large itemset is determined by concatenating items in the first large itemset and applying the minimum support test to the results. More iterations will take place until there are no more candidate itemsets. In simple terms, the cost of the algorithm is $O(I * D)$ where I denotes the number of iterations used. Association rules are generated based on all large itemsets. The generated rules are so large and complex that they can lead to conflicting results.

The Apriori algorithm is usually implemented on large data sets where the items within the one transaction are not in any particular order. This contradicts Web data sets where the pages are accessed in a particular order. Therefore, there was a need to implement sequential association rules using the Apriori algorithm. There are four types of sequential association rules presented by Yang et al. (2004):

1. Subsequence rules: they represent the sequential association rules where the items are listed in order.
2. Latest subsequence rules: They take into consideration the order of the items and most recent items in the set.
3. Substring rules: They take into consideration the order and the adjacency of the items.
4. Latest substring rules: They take into consideration the order of the items, the most recent items in the set as well as the adjacency of the items.

In this paper, we will use sequential association rule mining on user transaction data to discover Web page usage patterns. Prediction of the next page to be accessed by the user is performed by matching the discovered patterns against the user sessions. This is usually done online.

4 Proposed Model

4.1 Motivation for the Combined Approach

Our work is based on combining clustering algorithm, association rules mining and Markov model during the prediction process. The IPM integration during the prediction process is novel and proves to outperform each individual prediction model mentioned in section 1 as well as the different combination models addressed in section 2. The IPM integration model improves the prediction accuracy as opposed to other combinations that prove to improve the prediction coverage and complexity. The improvement in accuracy is based on different constraints like dividing the data set into a number of clusters based on services requested by users. This page categorization method proves to yield better clustering results (Wang et al. 2004). Therefore, better clusters means better Markov model prediction accuracy because the Markov model prediction will be based on more meaningfully grouped data. It also improves the state space complexity because Markov model prediction will be carried out on one particular cluster as opposed to the whole data set. The other constraint is using association rules mining in the case of a state absence in the training data or where the state prediction probability is not marginal. This helps improve the prediction accuracy because association rules look at more history and examine more states than Markov models. Also, IPM will not be subjected to the complexity associated with the number of rules generated because the rules will be examined in special cases only. Another constraint is the distance measure used in the identification of the appropriate cluster that each new page should belong to. The cosine distance measure has proved to outperform other distance measures like Euclidean, hamming, correlation and city block (Strehl et al. 2000, Halkidi et al. 2003). The prediction accuracy based on the integration of the three frameworks together according to these constraints proves to outperform the prediction accuracy based on each of the frameworks individually.

4.2 Algorithm

The process is as follows:

```
Training:

(1)Combine functionally related pages according
   to services requested

(2)Cluster user sessions into l-clusters

(3)Build a k-Markov model for each cluster

(4) For Markov model states where the
    majority is not clear

(5)  Discover association rules for each
     state

(6)EndFor
```

Combining similar pages or allocating related pages to categories is an important step in the training process of the IPM model. Consider a data set D containing N number of sessions. Let W be a user session including a sequence of pages visited by the user in a visit. $D = \{W_1, ..., W_N\}$. Let $P = \{p1, p2, \ldots, pm\}$ be a set of pages in a Web site. Since Markov model techniques will be implemented on the data, the pages have to remain in the order by which they were visited. $W_i = (p_1^i, ..., p_L^i)$ is a session of length L composed of multivariate feature vectors p. The set of pages P is divided into a number of categories C_i where $C_i = \{p1, p2, \ldots, pn\}$. This results in less number of pages since $C_i \subset P$ and $n < m$. For each session, a binary representation is used assuming each page is either visited or not visited. If the page is visited, a weight factor w is added to the pages representing the number of times the page was visited in the new session S_i. $S_i = \{(c_1^i, w_1^i), ..., (c_L^i, w_j^i)\}$. Ds is the data set containing N number of sessions S_N.

The categories are formed as follows:

Input: D containing N number of sessions W_N.

(1) For each page p_i in session W_i

(2) If $p_i \subset C_i$

(3) w_i.count++

(4) Else,

(5) $w_i = 0$

(6) EndIf

(7) EndFor

Output: Ds containing N number of Sessions S_N.

Combining the similar Web pages into categories Ci, increases the value of w and makes all sessions of equal length. According to Casale (2005), sessions of equal length give better similarity measures results. As an example, consider the following three sessions:

W1	1, 2, 3, 1, 3
W2	1, 2, 1,
W3	3, 1, 3

If pages 1 and 2 belong to category1 and page 3 belongs to category2, we have the following sessions:

Category	1	2
S1	3	2
S2	3	0
S3	1	2

Clustering the resulting sessions S_N was implemented using k-means clustering algorithm according to the Cosine distance between the sessions. Consider two sessions Sa and Sb. The Cosine distance between Sa and Sb is given by:

$$\text{distCosine}(Sa, Sb) = \frac{\sum(Sa_i Sb_i)}{\sqrt{\sum(Sa_i)^2}\sqrt{\sum(Sb_i)^2}} \quad (6)$$

Table 3 has 4 sessions with 4 pages each. If we are to form two clusters with two sessions each, we have to measure the distances between the sessions.

Table 3: Sessions

S1	3, 0, 5, 1
S2	2, 0, 5, 0
S3	0, 5, 0, 4
S4	0, 3, 0, 3

Table 4: Sessions distances

distCosine($S1, S2$)	0.019
distCosine($S1, S3$)	0.89
distCosine($S2, S3$)	1.0
distCosine($S1, S4$)	0.88
distCosine($S3, S4$)	0.06

Table 4 reveals the distances calculated using equation 1:

Clusters are formed according to the least distances between sessions, or the closest distances between sessions. Therefore, $\{S1, S2\}$ will form a cluster and $\{S3, S4\}$ will form another cluster.

```
Prediction:

(1)For each coming session

(2)   Find its closest cluster

(3)   Use corresponding Markov model to make
      prediction

(4)   If the predictions are made by states
      that do not belong to a majority class

(5)      Use association rules to make a
         revised prediction

(6)   EndIf

(7)EndFor
```

During the prediction process, each new page is examined and the appropriate cluster the new test point belongs to is identified. Let p_t be a new test point where $p_t \subset P$. Web sessions W are divided into K groups or clusters. The new point p_t has probability $prob(x_i = k)$ of belonging to cluster k where $\sum_k prob(x_i = k) = 1$ and x_i indicates the cluster membership of the new point p_t. The actual cluster k that the point p_t belongs to depends on the minimum distance of p_t to the mean values of K cluster centroids using the Cosine distance measure as follows:

$$\text{distCosine}(p_t, \mu) = \frac{\sum_{k=1}^{K}(p_t \mu)}{\sqrt{\sum_{k=1}^{K}(p_t)^2}\sqrt{\sum_{k=1}^{K}(\mu)^2}} \quad (7)$$

To continue with the prediction process, Markov model prediction is performed on the new identified cluster. If the Markov model prediction results in no state or a state that does not belong to the majority class, association rules mining is used instead. The majority class includes states with high probabilities where probability differences between two pages are significant. On the other hand, the minority class includes all other cases. In particular, the minority class includes:

1. States with high probabilities where probability differences between two pages are below (ϕ_c) or equal to zero.

2. States where test data does not match any of the Markov model outcomes.

A Markov model state is retained only if the probability difference between the most probable state and the second probable state is above (ϕ_c) (Deshpande & Karypis 2004). Another important issue here is defining the majority class and identifying whether the new state belongs to the majority or the minority class. This in mind, we employ the confidence pruned Markov model introduced by Deshpande *et. al*, (Deshpande & Karypis 2004). The confidence threshold is calculated as follows:

$$\phi_c = \hat{p} - z_{\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \quad (8)$$

Where $z_{\alpha/2}$ is the upper $\alpha/2$ percentage point of the standard normal distribution, and n is the frequency of the Markov state. Equation 5 stresses out the fact that states with high frequency would lead to smaller confidence threshold. That means that even if the difference between the two most probable pages is small, the state with higher probability will be chosen in the case of high frequency of the state occurrence. The smaller confidence threshold results in larger majority class. The effect of the confidence threshold value and, therefore, the majority class size on the prediction accuracy depends on the actual data set. To determine the optimal value of $z_{\alpha/2}$ and, as a result, the value of the confidence factor ϕ_c we conducted an experiment using the EPA data set (later referred to as D1 and described in section 6). As Table 5 depicts, the increase of the minority class or, in other words, the increase in the confidence factor is affected by the decrease of $z_{\alpha/2}$. During the prediction process, if the Markov model probability belongs to the minority class, association rules probability for the item is taken into consideration instead. Table 3 displays the results of the IPM accuracy using different values for $z_{\alpha/2}$. It is clear that the accuracy increases at first with lower confidence threshold and therefore, larger minority class. However, after a certain point, accuracy starts to decrease when the majority class is reduced to the extent where it looses the advantage of the accuracy obtained by combining Markov model and clustering. The optimal value for $z_{\alpha/2}$ is 1.15. Note that the number of states has dramatically decreased.

Table 5: Accuracy according to $z_{\alpha/2}$ value

$z_{\alpha/2}$	Accuracy	# states
0	31.29	9162
0.75	33.57	2061
0.84	35.45	1932
0.93	37.80	1744
1.03	40.60	1729
1.15	44.91	1706
1.28	43.81	1689
1.44	40.93	1614
1.64	38.85	1557
1.96	37.91	1479
2.57	36.81	1304

With $z_{\alpha/2}$=1.15, the most probable pages range approximately between 80% and 40% with ϕ_c ranging between 47% and zero respectively given n=2. This results in approximately 0.78 as the ratio of the majority class to the whole data set. This leaves space for 22% improvement using association rules mining not including instances that have zero matching states in the training data set.

4.3 Example

Consider table 6 that depicts data transactions performed by a user browsing a Web site.

Table 6: User sessions

T1	A,F,I,J,E,C,D,H,N,I,J,G,D,H,N,C,I,J,G
T2	F,D,H,N,I,J,E,A,C,D,H,N,I,J,G
T5	E,C,A,C,F,I,A,C,G,A,D,H,M,G,J
T3	F,D,H,I,J,E,H,F,I,J,E,D,H,M
T4	G,E,A,C,F,D,H,M,I,C,A,C,G

Performing clustering analysis on the data set using k-means clustering algorithm and Cosine distance measure where the number of clusters k=2 results in the following two clusters:

Cluster 1:

T1	A,F,I,J,E,C,D,H,N,I,J,G,D,H,N,C,I,J,G
T2	F,D,H,N,I,J,E,A,C,D,H,N,I,J,G
T3	F,D,H,I,J,E,H,F,I,J,E,D,H,M

Cluster 2:

T5	E,C,A,C,F,I,A,C,G,A,D,H,M,G,J
T4	G,E,A,C,F,D,H,M,I,C,A,C,G

Consider the following test data state I $\rightarrow$ J $\rightarrow$?. Applying the 2^{nd} order Markov Model to the above training user sessions we notice that the state $\langle I, J\rangle$ belongs to cluster 1 and it appeared 7 times as follows:

$$P_{l+1} = \text{argmax}\{P(E|J,I)\} = \text{argmax}\{E \rightarrow 0.57\}$$

$$P_{l+1} = \text{argmax}\{P(G|J,I)\} = \text{argmax}\{G \rightarrow 0.43\}$$

This information alone does not provide us with correct prediction of the next page to be accessed by the user as we have high probabilities for both pages, G and E. Although the result does not conclude with a tie, neither G nor E belong to the majority class. The difference between the two pages (0.14), is not higher than the confidence threshold (in this case 0.2745). In order to find out which page would lead to the most accurate prediction, we have to look at previous pages in history. This is where we use subsequence association rules as it appears in Table 7 below.

Table 7: User sessions history

A, F,	$\langle I, J\rangle$	E
C, D, H, N,	$\langle I, J\rangle$	G
D, H, N, C,	$\langle I, J\rangle$	G
F, D, H, N,	$\langle I, J\rangle$	E
A, C, D, H, N,	$\langle I, J\rangle$	G
F, D, H,	$\langle I, J\rangle$	E
H, F,	$\langle I, J\rangle$	E

Table 8 and Table 9 summarise the results of applying subsequence association rules to the training

Table 8: Confidence of accessing page E using subsequence association rules

A → E	AE/A	1/2	50%
F → E	FE/F	4/4	100%
D → E	DE/D	2/6	33%
H → E	HE/H	2/7	29%
N → E	NE/N	1/4	25%

Table 9: Confidence of accessing page G using subsequence association rules

C → G	CG/C	3/3	100%
D → G	DG/D	3/6	50%
H → G	HG/H	3/7	43%
N → G	NG/N	3/4	75%
A → G	AG/A	1/2	50%

data. Table 8 shows that F → E has the highest confidence of 100%. While Table 9 shows that C→ G has the highest confidence of 100%.

Using Markov models, we can determine that the next page to be accessed by the user after accessing the pages I and J could be either E or G. Whereas subsequence association rules take this result a step further by determining that if the user accesses page F before pages I and J, then there is a 100% confidence that the user will access page E next. Whereas, if the user visits page C before visiting pages I and J, then there is a 100% confidence that the user will access page G next.

5 Experimental Evaluation

In this section, we present experimental results to evaluate the performance of our algorithm. All experiments were conducted on a P4 1.8 GH PC with 1GB of RAM running Windows XP Professional. The algorithms were implemented using MATLAB.

For our experiments, the first step was to gather log files from active web servers. Usually, Web log files are the main source of data for any e-commerce or Web related session analysis (Spiliopoulou et al. 1999). The logs are an ASCII file with one line per request, with the following information: The host making the request, date and time of request, requested page, HTTP reply code and bytes in the reply. The first log file used is a day's worth of all HTTP requests to the EPA WWW server located at Research Triangle Park, NC. The logs were collected for Wednesday, August 30 1995. There were 47,748 total requests, 46,014 GET requests, 1,622 POST requests, 107 HEAD requests and 6 invalid requests. The second log file is SDSC-HTTP that contains a day's worth of all HTTP requests to the SDCS WWW server located at the San Diego Supercomputer Center in San Diego, California. The logs were collected from 00:00:00 PDT through 23:59:41 PDT on Tuesday, August 22 1995. There were 28,338 requests and no known losses. The third log file is CTI that contains a random sample of users visiting the CTI Web site for two weeks in April 2002. There were 115,460 total requests. The fourth log file is Saskatchewan-HTTP which contains one week worth of all HTTP requests to the University of Saskatchewan's WWW server. The log was collected from June 1, 1995 through June 7, 1995, a total of seven days. In this one week period there were 44,298 requests.

Before using the log files data, it was necessary to perform data preprocessing (Zhao et al. 2005, Sarukkai 2000). We removed erroneous and invalid pages. Those include HTTP error codes 400s, 500s, and HTTP 1.0 errors, as well as, 302 and 304 HTTP errors that involve requests with no server replies. We also eliminated multi-media files such as gif, jpg and script files such as js and cgi.

Next step was to identify user sessions. A session is a sequence of URLs requested by the same user within a reasonable time. The end of a session is determined by a 30 minute threshold between two consecutive web page requests. If the number of requests is more than the predefined threshold value, we conclude that the user is not a regular user; it is either a robot activity, a web spider or a programmed web crawler. The sessions of the data sets are of different lengths. They were represented by vectors with the number of occurrence of pages as weights.

Table 10 represents the different data sets after preprocessing.

Table 10: Sessions

	D1	D2	D3	D4
# Requests	47,748	28,338	115,460	44,298
# Sessions	2,520	4,356	13,745	5,673
# Pages	3,730	1,072	683	2,385
# Unique IPs	2,249	3,422	5,446	4,985

Further preprocessing of the Web log sessions took place by removing short sessions and only sessions with at least 5 pages were considered. This resulted in further reducing the number of sessions. Finally, sessions were categorized according to feature selection techniques introduced by Wang et al. (Wang et al. 2004). The pages were grouped according to services requested which yield best results if carried out according to functionality (Wang et al. 2004). This could be done either by removing the suffix of visited pages or the prefix. In our case, we could not merge according to suffix because, for example, pages with suffix index.html could mean any default page like OWOW/sec4/index.html or OWOW/sec9/index.html or ozone/index.html. Therefore, merging was according to a prefix. Since not all Web sites have a specific structure where we can go up the hierarchy to a suitable level, we had to come up with a suitable automatic method that can merge similar pages automatically. A program runs and examines each record. It only keeps the delimited and unique word. A manual examination of the results also takes place to further reduce the number of categories by combining similar pages.

5.1 Clustering, Markov Model and Association Rules

All clustering experiments were developed using MATLAB statistics toolbox. Since k-means computes different centroids each run and this yields different clustering results each time, the best clustering solution with the least sum of distances is considered using MATLAB k-means clustering solutions. Therefore, using Cosine distance measure with the number of clusters (k)=7 leads to good clustering results while keeping the number of clusters to a minimum.

Merging Web pages by web services according to functionality reduces the number of unique pages and, accordingly, the number of sessions. The categorized sessions were divided into 7 clusters using the k-means algorithm and according to the Cosine distance measure.

Markov model implementation was carried out for the original data in each cluster. The clusters were divided into a training set and a test set each and 2-Markov model accuracy was calculated accordingly. Then, using the test set, each transaction was considered as a new point and distance measures were calculated in order to define the cluster that the point belongs to. Next, 2-Markov model prediction accuracy was computed considering the transaction as a test set and only the cluster that the transaction belongs to as a training set.

Since association rules techniques require the determination of a minimum support factor and a confidence factor, we used the experimental data to help determine such factors. We can only consider rules with certain support factor and above a certain confidence threshold.

Using the D1, or EPA, data set, Figure 2 below shows that the number of generated association rules dramatically decreases with the increase of the minimum support threshold with a fixed 90% confidence factor. Reducing the confidence factor results in an increase in the number of rules generated. This is apparent in Figure 3 where the number of generated rules decreases with the increase of the confidence factor while the support threshold is a fixed 4% value. It is also apparent from Figure 2 and Figure 3 below that the influence of the minimum support factor is much greater on the number of rules than the influence of the confidence factor. The association rules precision is calculated as a fraction of correct recommendations to total test cases used.

$$Precision(Te) = \frac{Te \cap Tr}{Te} \tag{9}$$

Te represents the test cases whereas Tr represents training test cases or (D-Te).

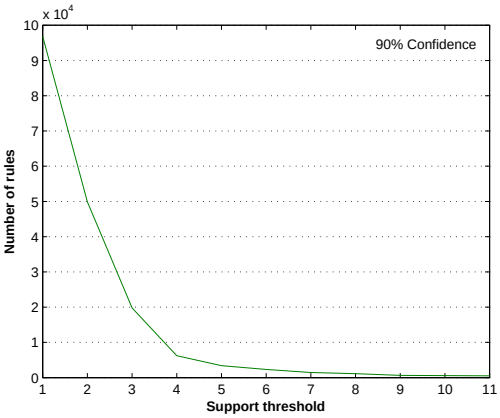

Figure 2: Number of rules generated according to different support threshold values and a fixed confidence factor: 90%.

Larger minimum support means less number of rules but it could also mean that genuine rules might be omitted. Figure 4 depicts the time complexity of generating association rules using different values of σ for D1 data set.

5.2 Experiments Results

Figure 5 depicts better Web page access prediction accuracy by integrating Markov model, Association rules and clustering (IPM). Prediction accuracy was computed as follows:

1. The data set is clustered according to k-means clustering algorithm and Cosine distance measure.

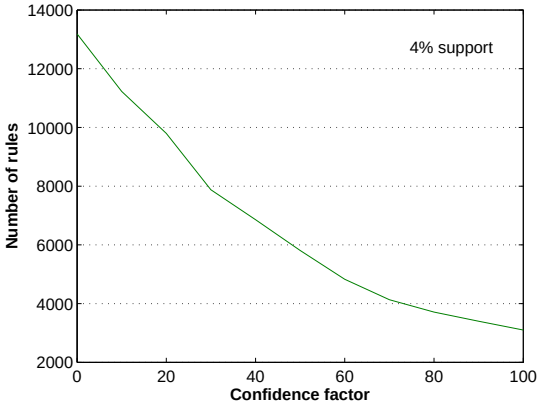

Figure 3: No. of rules generated according to a fixed support threshold: 4%.

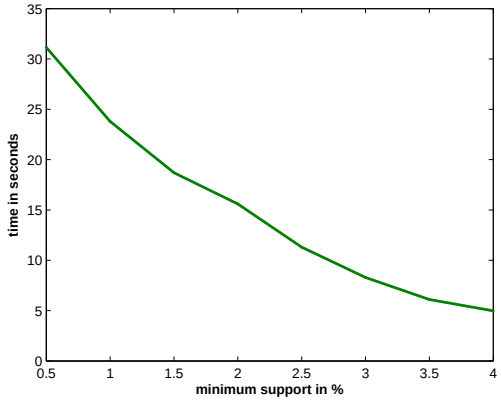

Figure 4: Time complexity in seconds for different support value.

2. For each new instance, the prediction accuracy is calculated based on the 2-MM performed on the closest cluster.
3. If the prediction results in a state that does not belong to the majority class, global association rules are used for prediction.
4. The frequency of the item is also determined in that particular cluster.
5. ϕ_c is calculated for the new instance using $z_{\alpha/2}$ value to determine if it belongs to the majority class.
6. if the state does not belong to the majority class, global association rules are used to determine the prediction accuracy, otherwise, the original accuracy is used.

Figure 5 shows that IPM results in better prediction accuracy than any of the other techniques individually. It also reveals that the increase in accuracy depends on the actual data set used. For instance, D4 prediction accuracy was increased while using a combination of MM and AR than by combining MM and clustering. On the other hand, D2 experienced more increase in accuracy using MM and clustering than using MM and AR. The accuracy increase of D1 and D3 was somewhat constant. Prediction accuracy results were achieved using the maximum likelihood based on conditional probabilities as stated in equation 4 above. All predictions in the test data that did not exist in the training data sets were assumed

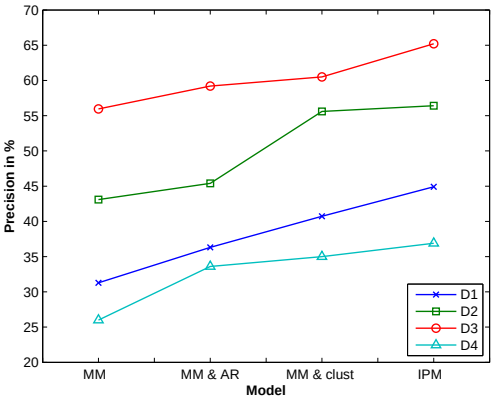

Figure 5: Precision of Markov model (MM) and MM with Association rules mining and MM with Clustering and all three models together (IPM) .

incorrect and were given a zero value. The Markov model accuracy was calculated using a 10-fold cross validation. The data was partitioned into T for testing and $(D - T)$ for training where D represents the data set. This procedure was repeated 10 times, each time T is moved by T number of transactions. The mean cross validation was evaluated as the average over the 10 runs. Table 11 reveals the standard deviation of all mean values of prediction accuracy for all four data sets.

Table 11: Accuracy values standard deviation

	D1	D2	D3	D4
MM	4.69	3.90	2.71	1.36
MM + AR	3.07	1.98	5.32	2.17
MM + Clust	2.55	2.94	1.45	3.83
IPM	1.32	3.07	6.19	2.69

The standard deviation results are considerably low compared to the mean values. This means that MM, MM + AR, MM + Clust and IPM accuracy results are quite different from each other lying on an improved baseline. The low standard deviation figures give more weight and significance to the improved prediction accuracy displayed in figure 5 above.

5.3 IPM Efficiency Analysis

All clustering runs were performed on a desktop PC with a Pentium IV Intel processor running at 2 GHz with 2 GB of RAM and 100 GB of hard disk memory. The runtime of the k-means algorithm, regardless of the distance measure used, is equivalent to O(nkl) (Jain et al. 1999), where n is the number of items, k is the number of clusters and l is the number of iterations taken by the algorithm to converge. For our experiments, where n and k are fixed, the algorithm has a linear time complexity in terms of the size of the data set. The k-means algorithm has a $O(k + n)$ space complexity. This is because it requires space to store the data matrix. It is feasible to store the data matrix in a secondary memory and then the space complexity will become O(k). k-means algorithm is more time and space efficient than hierarchical clustering algorithms with $O(n^2 logn)$ time complexity and $O(n^2)$ space complexity. As for all 2^{nd} order Markov model, the running time of the whole data set was similar to that of the clusters added together because the running time is in terms of the size of the data. i.e. T(n)=T(k1)+T(k2)+T(k3)+...T(ki) where time is denoted by T, the number of items in the data set is denoted by n, and the clusters are denoted by ki. The running time of association rule mining is O(I.D) as explained above. The association rules produced were for the whole data set. Accessing the appropriate rule is, however, performed online at time of prediction.

Constructing the IPM model is more complex than the individual models as it involves constructing k-means clustering, Markov model and association rules for the whole data sets. However, the IPM model prediction complexity is reduced due to the fact that the prediction process involves retrieving Markov models of one cluster as opposed to the whole data set. This reduces the running time by around 85%. Also, association rules are only retrieved in the case where the state does not belong to the majority class. This gives the conclusion that the complexity of IPM depends on the size of the majority class. Larger majority class yields less complex prediction as it involves less association rules accesses. However, larger majority class does not leave a larger room for accuracy improvement.

6 Conclusion

This paper improves the Web page access prediction accuracy by integrating all three prediction models: Markov model, Clustering and association rules according to certain constraints. Our model, IPM, integrates the three models using 2-Markov model computed on clusters achieved using k-means clustering algorithm and Cosine distance measures for states that belong to the majority class and performing association rules mining on the rest. The IPM model could be extended to a completely "hands-off" or automated system. Currently, some human intervention is required especially during the features selection process.

7 Acknowledgement

This work has been partially supported by ARC Discovery Grant DP0774450 to Li and Wang.

References

Adami, G., Avesani, P. & Sona, D. (2003), 'Clustering documents in a web directory', *WIDM'03, USA* pp. 66–73.

Agrawal, R. & Srikant, R. (1994), 'Fast algorithms for mining association rules', *VLDB'94, Chile* pp. 487–499.

Bouras, C. & Konidaris, A. (2004), 'Predictive prefetching on the web and its potential impact in the wide area', *WWW: Internet and Web Information Systems* (7), 143–179.

Cadez, I., Heckerman, D., Meek, C., Smyth, P. & White, S. (2003), 'Model-based clustering and visualization of navigation patterns on a web site', *Data Mining and Knowledge Discovery* **7**.

Casale, G. (2005), 'Combining queueing networks and web usage mining techniques for web performance analysis', *ACM Symposium on Applied Computing* pp. 1699–1703.

chen, M., LaPaugh, A. S. & Singh, J. P. (2002), 'Predicting category accesses for a user in a structured information space', *SIGIR'02, Finland* pp. 65–72.

Deshpande, M. & Karypis, G. (2004), 'Selective markov models for predicting web page accesses', *Transactions on Internet Technology* **4**(2), 163–184.

Eick, C. F., Zeidat, N. & Zhao, Z. (2004), 'Supervised clustering - algorithms and benefits', *IEEE ICTAI'04* pp. 774–776.

Eirinaki, M., Vazirgiannis, M. & Kapogiannis, D. (2005), 'Web path recommendations based on page ranking and markov models', *WIDM'05* pp. 2–9.

Halkidi, M., Nguyen, B., Varlamis, I. & Vazirgiannis, M. (2003), 'Thesus: Organizing web document collections based on link semantics', *The VLDB Journal* **2003**(12), 320–332.

Jain, A. K., Murty, M. N. & Flynn, P. J. (1999), 'Data clustering: A review', *ACM Computing Surveys* **31**(3), 264–323.

Kim, D., Adam, N., Alturi, V., Bieber, M. & Yesha, Y. (2004), 'A clickstream-based collaborative filtering personalization model: Towards a better performance', *WIDM '04* pp. 88–95.

Lai, H. & Yang, T. C. (2000), 'A group-based inference approach to customized marketing on the web - integrating clustering and association rules techniques', *Hawaii International Conference on System Sciences* pp. 37–46.

Liu, F., Lu, Z. & Lu, S. (2001), 'Mining association rules using clustering', *Intelligent Data Analysis* (5), 309–326.

Lu, L., Dunham, M. & Meng, Y. (2005), 'Discovery of significant usage patterns from clusters of clickstream data', *WebKDD '05* .

Mathur, V. & Apte, V. (2007), 'An overhead and resource contention aware analytical model for overloaded web servers', *WOSP'07, Argentina* .

Mobasher, B., Dai, H., Luo, T. & Nakagawa, M. (2001), 'Effective personalization based on association rule discovery from web usage data', *WIDM'01, USA* pp. 9–15.

Papadakis, N. K. & Skoutas, D. (2005), 'STAVIES: A system for information extraction from unknown web data sources through automatic web warpper generation using clustering techniques', *IEEE Transactions on Knowledge and Data Engineering* **17**(12), 1638–1652.

Pitkow, J. & Pirolli, P. (1999), 'Mining longest repeating subsequences to predict www surfing', *USENIX Annual Technical Conference* pp. 139–150.

Pons, A. P. (2006), 'Object prefetching using semantic links', *The DATA BASE for Advances in Information Systems* **37**(1), 97–109.

Rigou, M., Sirmakesses, S. & Tzimas, G. (2006), 'A method for personalized clustering in data intensive web applications', *APS'06, Denmark* pp. 35–40.

Sarukkai, R. (2000), 'Link prediction and path analysis using markov chains', *9th International WWW Conference, Amsterdam* pp. 377–386.

Spiliopoulou, M., Faulstich, L. C. & Winkler, K. (1999), 'A data miner analysing the navigational behaviour of web users', *Workshop on Machine Learning in User Modelling of the ACAI'99, Greece* .

Srivastava, J., Cooley, R., Deshpande, M. & Tan, P. (2000), 'Web usage mining: Discovery and applications of usage patterns from web data.', *SIGDD Explorations* **1**(2), 12–23.

Strehl, A., Ghosh, J. & Mooney, R. J. (2000), 'Impact of similarity measures on web-page clustering', *AI for Web Search* pp. 58–64.

Wang, Q., Makaroff, D. J. & Edwards, H. K. (2004), 'Characterizing customer groups for an e-commerce website', *EC'04, USA* pp. 218–227.

Yang, Q., Li, T. & Wang, K. (2004), 'Building association-rule based sequential classifiers for web-document prediction', *Journal of Data Mining and Knowledge Discovery* **8**.

Yong, W., Zhanhuai, L. & Yang, Z. (2005), 'Mining sequential association-rule for improving web document prediction', *ICCIMA'05* pp. 146–151.

Zhao, Q., Bhomick, S. S. & Gruenwald, L. (2005), 'Wam miner: In the search of web access motifs from historical web log data', *CIKM'05, Germany* pp. 421–428.

Zhong, S. & Ghosh, J. (2003), 'A unified framework for model-based clustering', *Machine Learning Research* **4**, 1001–1037.

Zhu, J., Hong, J. & Hughes, J. G. (2002), 'Using markov models for web site link prediction', *HT'02, USA* pp. 169–170.