

Web Server Performance Optimization using Prediction Prefetching Engine

Silky Makker

Dept of Computer Science & Engg,
Manav Rachna International University
Faridabad, India

R.K Rathy

Dept of Computer Science & Engg,
Manav Rachna International University
Faridabad, India

ABSTRACT

The long-term success of the World Wide Web depends on fast response time. People use the Web to access information from remote sites, but do not like to wait long for their results. The rapid growth in the amount of information and the number of users has lead to difficulty in providing effective response time for the web users and this increased web latency; resulting in decreased web performance. Although several proposals have been made for reducing this latency, like it can be improved by caching, the benefit of using it is rather limited owing to filling the cache with documents without any prior knowledge. Predictive caching becomes an attractive solution wherein the forthcoming page likely to be requested soon are predicted based on user access logs information and pre-fetched, while the user is browsing the current display pages. As web page prediction gained its importance, This paper proposes a bracing approach for increasing web server performance by analyzing user behavior, in this pre-fetching and prediction is done by pre-processing the user access log and integrating the three techniques i.e. Clustering, Markov model and association rules which achieves better web page access prediction accuracy; This work also overcomes the limitation of path completion i.e. by extracting web site structure paths are completed, which helps in better prediction, decreasing access time of user and improving web performance.

Keywords

Web log file, pattern discovery, Petri nets, pre-fetching, prediction Association rules, Markov models, clustering.

1. INTRODUCTION

Web users are facing the problems of information overload and drowning due to the significant and rapid growth in the amount of information and the number of users. The size of publically indexed World Wide Web has probably surpassed 24.39 billion pages in June 2010, As a result, Delays in access to Web based Information continues to be a serious problem even with higher network bandwidth, due to overhead web latency has increased due to which web performance has decreased. User perceived latency s from several sources such as bandwidth, speed, overhead, accessing the web page etc. In accordance of "Eight Second Rule"[1], it can be observed that web latency affects the user work and lot of efforts are taken to minimize the latency perceived by the user. Caching of web documents has been developed to reduce the latency but it has the drawback that it stores the pages without any prior knowledge i.e. the hit ratio is less. While predictive caching can optimize the www in many respects, this motivates our work, in this paper Web page predictive caching is done by using Prediction processes like clustering, association rules, and Markov model[2], so as to reduce the accessing time of user or disk access ultimately

improve web server performance. Predictive caching is the speculative retrieval of a resource into a cache based on user access log; in the anticipation that it can be served from cache in the future[3] leading to improvement in web server performance.

Web page access prediction gained its importance from the ever increasing number of e-commerce and e-businesses[4]. It involves personalizing, Marketing, Recommendations, helps in improving the web site structure and also guide web users in navigating through hyperlinks for accessing the information they need. The most widely used techniques for discovering the patterns are Markov model, association rules and clustering, sequential patterns etc. However, each of the aforementioned techniques has its own limitations, especially when it comes to accuracy and space complexity[5]. In proposed work pre-fetching and prediction is done by pre-processing of logs as it is the main requirement to provide user with best recommendations[6] and also overcomes the limitation of path completion and for pattern dicover we integrate the following three techniques together i.e. clustering, association rules and low-order Markov model using frequency support pruning, it achieves complete logs, better accuracy, less state space complexity and less number of rules. The predicted pages are pre-fetched and keep it in server cache which reduces the accessing time of that page and increases the web server performance.

This paper is organized in the following way. Section 2 discusses related work done in this domain. Architecture of the proposed system is presented in Section 3. Sequence of steps of the work is discussed in Section 4. Finally Section 5 comprise of the conclusion.

2. REVIEW

Padmanabhan et al [7] investigated ways of optimizing retrieval latency. Web caching has been recognized as an effective solution to minimize user access latency A method of called prefetching was introduced in which clients in collaboration with servers prefetch web page that the user is likely to access soon, while he/she is viewing the currently displayed pagel [8]. The benefit of prefetching is to provide low retrieval latency for users, which can be explained as high hit ratio Investigate an approach to reduce web latency by prefetching between caching, proxies, and browsers. Research on predictive Web prefetching has involved the important issue of log file processing and the determination of user transactions (sessions) from it [6,9,10,11,12] provide various data mining algorithms for the path traversal patterns and how to efficiently mine the access patterns from the web logs. Pirolli and Pitkov [11] predict the next web page by discovering the longest repeating subsequence in the web sessions. Liu et al [12] used association rules for web access predicions. Yang et al

[13] studied different association rule based methods for web request prediction. Using association rules for web 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. Padbanabham and Mogul [14] use N-hop Markov models predicted the next web page users will most likely access by matching the user's current access sequence with the user's historical web access sequences for improving prefetching strategies for web caches. Sarukkai [15] used first-order Markov models to model the sequence of pages requested by a user for predicting the next page accessed. Liu et al [16] integrated association rules and clustering for reducing the overhead associated with large database. Cadez et al [17] integrated clustering and first order Markov model to increase accuracy. Kim et al [18] combined association rules and Markov model for web access prediction. In this paper, Integrated data mining techniques are applied for extracting user access patterns from web access logs. Further this work uses application of petri nets (PN) for path completion which enhance web usage mining. Web structure is extracted using parsing algorithm, from which incidence matrix is built. The web structure information in the incidence matrix and the reachability properties obtained from the PN model help in path competition process.

Several researchers, including our work, attempted to improve the Web page access prediction precision or coverage by combining clustering with association rules Liu et al [12,13] 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 rule techniques on each cluster. 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) [19]. It 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) [19] work improved recall and it did not improve the Web page prediction accuracy.

The Integrated Prediction Model is novel and proves to outperform each individual prediction model as well as the different combination models addressed above. 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) [20]. 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 rule 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 subject to the complexity associated with the number of rules generated because the rules will be examined in special cases only.

2.1 Markov Model

Markov models [5] are becoming very commonly 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 et al. 2004) [21]. Let $P = \{p_1, p_2, \dots, p_m\}$ 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 $\text{prob}(p_i|W)$ is the probability that the user visits pages p_i next. Page p_{l+1} the user will visit next is estimated by:

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

This probability, $\text{prob}(p_i|W)$, is estimated by using all W sequences of all users in history (or training data), denoted by W . Naturally, the longer l and the larger W , the more accurate $\text{prob}(p_i|W)$. However, it is infeasible to have very long l and large W and it leads to unnecessary complexity. Therefore, to overcome this is estimated by assuming that the sequence of the Web pages visited by users follows a Markov process. The Markov process imposed 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 $k \ll l$. The equation becomes:

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

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 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.

In particular, three schemes are used for pruning the states of the All- k th-Order Markov model, called support frequency pruning, confidence pruning and error pruning. For the proposed work frequency pruning is used, 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 as many states have low predictive reliability since their occurrence frequencies are very low. So these low frequency states are removed as they affects the accuracy of a Markov model. The evaluation of the pruning has shown that up to 90% of the states can be pruned leading to less state space complexity and increased coverage but accuracy remains unchanged. Although the all- k th order models solve the reduced accuracy problem, they give rise to another major problem, the state space complexity. As the order increases the states increases so it is used in combination with association or clustering.

2.2 Association Rules

Association rule mining is a major pattern discovery technique [13,14] The patterns are discovered based on previous browsing

history. The goal of association rule mining is to solve market basket Analysis but the applications of association rules are far beyond that.. The main disadvantage of association rule mining is that many rules are generated, which results in contradictory predictions for a user session. Another problem associated with it is the frequent item problem where the items that occur together with a high frequency will also appear together in many of the resulting rules and, thus, resulting in inconsistent predictions. As a consequence, a system cannot give recommendations when the data set is large.

An implication is called an association rule if its support and confidence are not less than some user specified minimum thresholds. The minimum support requirement dictates the efficiency of association rule mining. One major motive for using the support factor comes from the fact that we are usually interested only in rules with certain popularity. Support corresponds to statistical significance, and confidence is a measure of the rules strength. A rule with a small support measure will have a higher observed error rate with the same number of wrong predictions. However, a higher support measure will face the complications of missing some useful rules.

$$support(X \Rightarrow Y) = P(X \cup Y) \quad (3)$$

$$confidence(X \Rightarrow Y) = P(Y|X) \quad (4)$$

In relevance to Web usage mining, the discovery of association rules usually aims at the discovery of associations or relation between Web pages based on their co-occurrence in user sessions Mobasher et al. (1999)[9]. Using association rules for web page access prediction involves a contradictory results as they generate too many rules. So a combination of Association rules and Markov models are most commonly used approach for this type of predictions. A low order Markov model provides high coverage with low state space complexity, and association rules help achieve better accuracy.

For the purposed work, subsequence sequential association rule mining is used on user transaction data to discover Web page usage patterns, they represent the rules where the items are listed in order ,whereas there are other sequential association rules also like Latest subsequence rules, Substring rules, Latest substring rules. The immense number of generated rules gives rise to the need of some predictive models that reduce the rule numbers and increase their quality by weeding out the rules that were never applied. The predictive model we have used is Most-confidence matching in this the rule with the highest confidence is chosen amongst the rest of all the applicable rules whose support values are above a certain threshold, there are other models also like Longest match, Least error matching.

2.3 Clustering

Clustering can be considered the most important *unsupervised learning* problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data [22]. The goal of clustering is to determine the intrinsic grouping in a set of unlabeled data. There are broadly two grouping schemes: hierarchical and partitional schemes. The hierarchical schemes are more versatile, and the partitional schemes are less expensive. The partitional algorithms aim at maximizing the squared error criterion function. Motivated by the failure of the squared error partitional clustering algorithms in finding the optimal solution to this problem, a large collection of approaches have been proposed and used to obtain the global optimal solution to this problem. In

this we have used K-means algorithm using session based similarity measures for clustering the user session into k-clusters. If two users accessed the same pages in sessions, they might have some similar interests in the sense that they are interested in the same information. The number of common pages they accessed can measure this similarity. The measure is defined by using the equation mentioned

$$sim1(pi, sj) = \frac{\sum_k(use(pk,si)*use(pk,sj))}{\sqrt{\sum_k(use(pk,si)*\sum_k(use(pk,sj))}} \quad (5)$$

where $\sum_k(use(pk,si))$ is the total number of pages that were accessed by the user of session s and $\sqrt{\sum_k(use(pk,si) * \sum_k(use(pk,sj))}$ is the number of common pages accessed by both s , and s By Assuming $k=2$ and two of the user sessions as center

By using the similarity matrix we cluster the user sessions which are highly similar to each other. So by using the clustering better prediction is performed as we have to apply prediction algorithm to a specific cluster rather than whole data set.

3. PROPOSED ARCHITECTURE

This section describes the architecture of the proposed system shown in Figure 1. Following subsection describes various components of the proposed system.

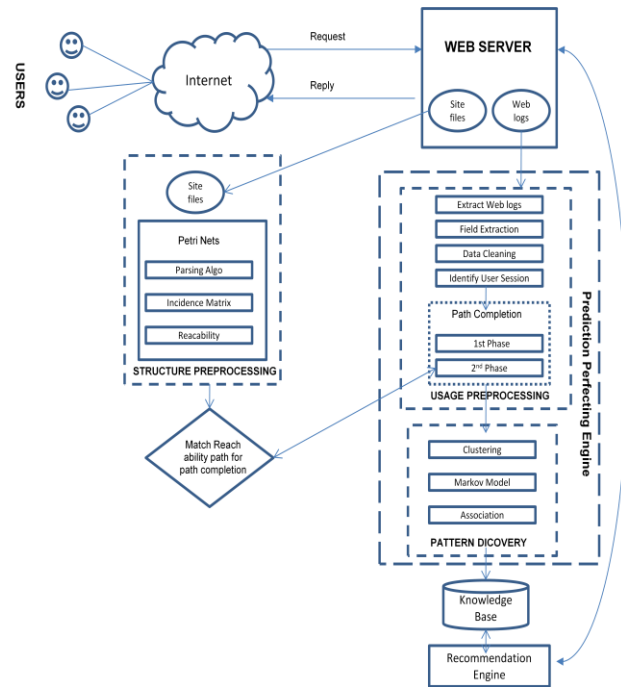


Figure1. Proposed System Architecture

3.1 Web Server

The primary function of the web server is to deliver web pages to the users. The records of all the users/clients that send requests to the server are kept on the web logs, which reside at the server side. This log file helps in analyzing user access patterns and in predicting next page likely to be accessed by the web user. It also

contains site files i.e. hyperlink of web pages of a particular web site.

3.2 Prediction Pre-fetching Engine (PPE)

Prediction Pre-fetching Engine (PPE) [20] processes the past references, tracking the user behavior to deduce the probability of future access of the web page based on access log information.

The Predictive Pre-fetching Engine which sits behind the search engine Interface. It comprises of two parts usage preprocessing and pattern discovery phase, discussed in following subsections.

3.2.1 Usage Pre preprocessing

Data preprocessing involves field extraction, data cleaning, user session identification and path completion.

3.2.1.1 Field Extraction

The very important task which is required in all the preprocessing phase is known as field extraction i.e. separating various fields from the single line of the logs file into a form of table by using space as field separator.

3.2.1.2 Data cleaning

Cleaning the web log file involves removal of irrelevant items like the image file (GIF and JPEG) and java script files (JS), style sheet (.CSS), the information search by agents i.e. robot.txt, bot, slurp etc., and the status code which are which are not successful in providing web page i.e. below 200 to above 299 as these do not contribute for the patterns relevance.

3.2.1.3 User session Identification

A user session is defined as the sequence of requests made by the single end user during a visit to a particular site. For different users unique IP Address is considered and within a single session, a user may follow links to several pages that belong to the similar pattern but during the same session, it may also be possible that the user might visit some other pages that do not belong to the same pattern i.e. user session may contain the documents belonging to patterns while others that do not and the documents are interleaved in the session. For session division timeout taken was 30 minutes.

3.2.1.4 Path Completion

As due to caching and proxy servers the information in the web logs are incomplete, so for path completion two phases are used.

3.2.1.4.1 1st phase: Referrer Information

The first phase of the algorithm uses the referrer information to restore the missing click stream data. For the first click stream record in session check if it has any referrer information. If it does, and if the referrer belongs to the current site, some click stream data must have been lost and needs to be restored. The restoration rule uses the referrer information of the first record as the URL of a new record. The referrer information of this new record is set to a null value, the time stamp is set to a suitable time (in this case 5 seconds earlier than the first record) and the record is marked as "Restored" and if other than 1st record is there then check referrer with previous url, if different then data must be resorted.

3.2.1.4.2 2nd phase: Web site link Structure

The second phase of the algorithm uses information from the website's link structure to insert the "most probable" browsing path of the user. To do this, a list of pages and the link structure for the site must be maintained. So for making the website structure we use the application called Petri Nets of structure preprocessing.

3.3 Structure Preprocessing

In this site files are extracted from the web server to find the hyperlink of website.

3.3.1 Petri nets

Petri Nets (PN) [23] is a high-level graphical model widely used to find out the reach ability paths. PN can store the analyzed results in a matrix for future follow-up analyses. PN defines directed arcs from places to transitions, where N is a set of nonnegative integers;

an output function that defines directed arcs from transitions to places.

3.3.1.1 Parsing Algorithm

The execution of parsing algorithm help to find out the number of transition taken place and the incoming and outgoing lines from place to transition, basically it helps to make the incidence matrix.

3.3.1.2 Incidence Matrix

An incidence matrix records all token-amount changes in all places after all fired transitions. For PN with n transitions and m places, the incidence matrix A, where $A=[a_{ij}]$, is an $n \times m$ matrix of integers; its typical entry is given by $a_{ij} = a_{ij+} - a_{ij-}$ where $a_{ij+} = O(t_i, p_j)$, the weight of the arc from Transition i to its Output Place j, and $a_{ij-} = I(t_i, p_j)$, the weight of the arc to Transition i from its Input Place j; a_{ij+} , a_{ij-} and a_{ij} represent the number of tokens removed, added, and changed in Place p_j , respectively, when Transition t_i fires once.

3.3.1.3 Reachability Paths

So from Incidence matrix reach ability paths can be found out which helps in path completion.

After all the preprocessing is done, the cleaner version of the logs is formed which act as a input or database on which various data mining techniques are applied for generating the rules.

3.4 Pattern Discovery

Data mining is the process of extracting hidden patterns from data. After the preprocessing of the web server log file, data mining techniques are applied. In this work, combining the three mining techniques clustering, Markov model and association rules together for prediction is proposed. It first clusters Web sessions according to meaningful features selection techniques using k-means clustering algorithm and similarity measure. Each data set is grouped into different number of clusters. The integration model then computes Markov model for prediction on the resulting clusters, and if there is ambiguity then Association rules are applied for accurate results.

3.5 Knowledge Base

The knowledge base is a repository of extracted rules which have been derived by applying data mining techniques. The attributes it contains are Number of users, the web page they access and the time of accessing the web pages.

3.6 Recommendation Engine

Whenever user request for URL, Prediction Pre-fetching Engine (PPE) sends the URL to the recommendation engine, which in turn does prediction based on rules from knowledge base. Finally based on this, it pre-fetch the web page in server cache before he/she explicitly request for, thus decreasing access time of retrieving that page and improving web server performance.

4. ALGORITHM

This Section discusses the sequence of steps of the work done by the proposed system.

4.1 Algorithm: UpdateWhenNewQuery($qi+1$)

1. Preprocess the log file by following steps
2. Field extraction by using space as a field separator
3. data cleaning by removing the records having keywords(.jpg, .jpeg, .gif, robots.txt, slurp, bot, script, .css) in URL and status code <200 &>299.
4. user session identification
 add first entry to user i;
 For each (next entry)
 begin
 if(IP == Previous IP)
 if(time[this entry] - time[this entry -1]
 add entry to user session i
 path completion IF Referrer 1 of user
 session (i) is not null
5. Then some data is missing and must be restored
6. if referrer 1 belongs to website
7. then Restored before that record as S1' [IP1, T1-5, Referrer1, -, Restored] end if
 for i=2 to n
 if Li - 1 ≠ Referrer i
 then Restore Si' [IP1, (Ti-1+Ti)/2,
 Referrer i, -, Restored], end for
 if url (i) has no direct link to url(i+1)
 then complete path by matching with reach
 ability paths obtained through the application
 of Petri nets
8. After completing user sessions
9. Combine functionally related pages according to services requested
10. Cluster user sessions into l-clusters using k-means using similarity measures
11. Using Frequency Markov model for pruning the states build a k-Markov model for each cluster
12. for Markov model states where the majority is not clear
13. Collect all sessions satisfying the state
14. Construct association rules to resolve ambiguity
15. Store the association rules with the state
16. End for
17. Discover Patterns(qlogi+1) // will return rules
18. KB ← rules
19. End for
20. End

4.2. Algorithm: Recommendation Engine ($qi+1$)

1. begin
2. for each new query or For each coming user session
3. make predictions based on KB
4. Otherwise find its closest cluster
5. Use corresponding Markov model to make prediction
6. If the predictions are made by states that do not belong to a majority class
7. Use association rules to make a revised predictions
8. End If
9. Pre fetch pages
10. End For
11. End For
12. End

5. CONCLUSION AND FUTURE WORK

The main objective of this work is to help achieve better prediction accuracy for Web page access. The main technology implemented for this purpose is through using Web usage mining pattern discovery techniques. This paper proposes a novel approach for predicting user behavior for improving web performance. In this prediction and pre-fetching is done by collecting information from user access log. This work overcomes the limitation of path completion. Application of Petri Nets for extracting web site structure helps in path completion process, better prediction, decreasing web latency and improving web performance. We first discussed major issues related to each of the pattern discovery techniques in general. We then identified their limitations and integrated them differently in a way where those limitations are addressed properly and kept to a minimum. Through the pattern discovery models integration, we exhausted their varied positive impact on Web page prediction accuracy. By keeping the models limitations to a minimum and relying on their advantages, and, by integrating the different models according to different constraints, we were able to achieve more accurate prediction results.

In this paper we have proposed the improved Web page access prediction accuracy by combining different Web usage mining techniques. First we have examined the individual Web usage mining techniques individually and demonstrated experimentally the fact that Web access prediction accuracy increases in this order of using Web usage mining techniques by the combination of association rules, clustering and Markov model. As an end result, through our integration of Markov model, association rule mining and clustering together, we have proved to increase the Web access prediction accuracy significantly. The extra advantage of our models is the low state space complexity. Combining a lower order Markov model with association rules, and with clustering, and finally, with both association rules and clustering has benefited from the low order Markov model low state space complexity. All integration models implemented proved to generated less states than a higher order Markov model.

6. REFERENCES

- [1] Network performance, available at http://en.wikipedia.org/wiki/Network_performance
- [2] Pallis, G., Vakali, A., Pokorny, J: A Clustering-based Prefetching Scheme on a Web Cache Environment. *Int. Journal Computers & Electrical Engineering*, Elsevier, Vol. 34, Issue 4, pp. 309-323
- [3] Venkata N. Padmanabhan. "Improving World Wide Web Latency", Technical Report, Computer Science Division, University of California, Berkeley, CA, May, 1995.
- [4] Khalil, F., Li, J. and Wang, H. (2007) 'Integrating Markov model with clustering for predicting web page accesses', *Australian World Wide Web (AusWeb'07)*, Coffs Harbour, Australia, pp.63-74.
- [5] Khalil, Faten (2008) Combining web data mining techniques for web page access prediction. [Thesis (_PhD/Research)]
- [6] Cooley, R., Mobasher, B., and Srivastava, J. Data preparation for mining World Wide Web browsing patterns. *Journal of Knowledge and Information Systems* 1, 1 (1999).
- [7] Venkata N. Padmanabhan. "Improving World Wide Web Latency", Technical Report, Computer Science Division, University of California, Berkeley, CA, May, 1995
- [8] Fan L., Cao P., and Jacobson Q., "Web prefetching between Low-Bandwidth Clients and proxies: potential and performance." In *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems.*, May 1999.
- [9] P. Atzeni, G. Mecca, and P. Merialdo, "To Weave the Web," *Proc. 23rd Conf. Very Large Data Bases (VLDB '97)*, pp. 206-215, Aug.1997,
- [10] Payal Gulati, A.K. Sharma, Amit Goel, Jyoti Pandey, "A Novel Approach for Determining Next Page Access," *icetet*, pp.1109-1113, 2008 First International Conference on Emerging Trends in Engineering and Technology, 2008.
- [11] Pitkov, J. and Piroli, P. "Mining Longest repeating Subsequences to predict world wide web surfing, *Proc. USENIX Symp. On Internet Technologies and Systems*, 1999.
- [12] Liu, B., Hsu, W., and Ma, Y., "Integrating Classification and Association Mining", *Proc. of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*, 1998.
- [13] Yang, Q., Li, T., Wang, K., "Building Association Rules Based Sequential Classifiers for Web Document Prediction", *journal of Data Mining and Knowledge Discovery*, Netherland: Kluwer Academic Publisher, 2004.
- [14] V. Padmanabhan and J. Mogul, "Using Predictive prefetching to improve World Wide Web latency", *ACM SIGCOMM Computer Comm. Rev.*, Vol. 26,no.3, July 1996.
- [15] Sarukkai, R.R., "Link prediction and path analysis using Markov chain", *proc. of the 9th International World Wide Web Conference on Computer networks*, 2000.
- [16] Liu, F. Lu Z. "Mining Association rules using clustering", *Intelligent Data Analysis*, 2001.
- [17] Cadez I., Heckerman D., Meek C. Symth P., and Whire S., "Visualization of Navigation Patterns on a website using Model Based Clustering", March, 2002
- [18] Kim, D., Adam, N. Alturi, V., Bieber, M. & Yesha, Y. "A clickstream - based collaborative filtering personalization model: Towards a better performance", *WIDM*, 2004
- [19] Po-Zung Chen, Chu-Hao Sun, Shih-Yang Yang, "Modeling and Analysis the Web Structure Using Stochastic Timed Petri Nets", *Journal of Software*, Vol. 3, No. 8, November 2008.
- [20] Payal Gulati, A.K. Sharma, Amit Goel, Jyoti Pandey, "A Novel Approach for Determining Next Page Access," *icetet*, pp.1109-1113, 2008 First International Conference on Emerging Trends in Engineering and Technology, 2008.
- [21] Deshpande, M. and G. Karypis, 2004. Selective markov models for predicting web page accesses. *ACM Transact. Internet Technol.*, 4: 163-184.
- [22] R. Ng and J. Han, "CLARANS: A Method for Clustering Objects for Spatial Data Mining," *IEEE Trans. Knowledge and Data Eng.*, vol. 14, no. 5, pp. 1003-1016, Sept./Oct. 2002.
- [23] Jiacun Wang: *Timed Petri Nets, Theory and Application*, Boston: Kluwer Academic Publishers, 1998.