# A Weighted Markov Model for Web Pre-fetching to Improve User Interface over Internet

Ms. Veena Singh Bhadauriya[1], Dr. Bhupesh Gour[2], Dr. Asif Ullah Khan[3]
Department of Computer Science & Engineering[1,2,3]
TIT, Bhopal[1,2,3]
[1]veena2003@yahoo.com

**-------------------------------------------------------------------ABSTRACT------------------------------------------------------------**

**Rapid growth of web application has increased the researcher's interests in this era. All over the world has surrounded by the computer network. There is a very useful application call web application used for the communication and data transfer. An application that is accessed via a web browser over a network is called the web application. Web caching is a well-known strategy for improving the performance of Web based system by keeping Web objects that are likely to be used in the near future in location closer to user. The Web caching mechanisms are implemented at three levels: client level, proxy level and original server level. Significantly, proxy servers play the key roles between users and web sites in lessening of the response time of user requests and saving of network bandwidth. Therefore, for achieving better response time, an efficient caching approach should be built in a proxy server. This paper use FP growth, weighted rule mining concept and Markov model for fast and frequent web pre fetching in order to has improved the hit ratio of the web page and expedites users visiting speed.**

**Keywords: Web Services, Pre-fetching, Log file**

Date of Submission: October 06, 2013 Date of Acceptance: December 12, 2013

## I. INTRODUCTION

With the growth of the internet, use of Web sites across the world has been increased rapidity. where each website having number of web pages, number of indexed Web pages of over 20 billion. With such a huge collection, a quick Web search and response to user's query is essential for effective utilization of the Web.

Number of hardware and software research has been done in order to increases efficiency of Web servers, where hardware solution includes improvement in speed, bandwidth and software solutions encompasses more suitable protocols model and algorithms.

More suitable software solution is per-fetching that preloads some relevant data to local machine on the basis of probability calculation from historical information ie past records available in web server log file. Web pre-fetching means, on the basis of past web accessing behavior of a client deduct of upcoming page quickly available on locally site rather than retrieved from remote sites. Obviously page retrieval in preloading process is from remote sources, but it can be done in advance and serve when needed without recognized delay from the user's prospective, all pre-fetching technique used time gap between consecutive requests from the same user in the Web environment and the Web server can use this time gap to pre-fetch the predicted pages. Successful perfecting will not only reduce the delays for users' requests for Web objects, but also result in less overall network traffic and lighter loads on the Web servers.

Number of idea has been proposed by various authors in recent year. This paper demonstrate the frequent mining pattern which is obtain on the basis of input and on the basis of that caching and pre-fetching ratio is calculated. Thus we present a new idea for the interpretation of Web pre-fetching and web caching from the given usage items that encapsulate FP growth [2], weighted rule mining concept [] and Markov model[16] gives much better performance than the other ones, in the quantitative measures such as hit ratios and byte hit ratios of accessed information.

The main motive of this paper is to design a pre-fetching and caching ratio model used to improve hit ratios of accessed documents, the architecture of which consists of three functional a mining mechanism consisting of the pattern mining, here in this frequent item set is found by using graph approach and based on its frequent pattern are discover, based on this caching and pre-fetching ratio is found out and Finally make conclusions.

This paper is divided into seven sections. First one is introduction in which give the brief description of work. The second section discusses the previous work related to the topic. The third section describes the approach used in the presented work. The next section describes the proposed architecture of the presented work. After this the simulation result has discussed. Finally paper concludes in the section eight's.

## II. PREVIOUS WORK

Research over web mining and web pre-fetching is going very fast in last decades. Toufiq Hossain Kazi et.al [11] gives an Adaptive Resonance Theory (ART) based on pre-fetch technique namely ART1, use the bottom-up and top-down weights of the cluster-URL connections obtained from a modified ART1 algorithm to make pre-fetching decisions. A.B.M.Rezbaul Islam et.al [21] proposed a new and improved FP tree with a table and a new algorithm for mining association rules. This algorithm mines all possible frequent item set without generating the conditional FP tree. It also provides the frequency of frequent items, which is used to estimate the desired association rules, Whereas P. Sampath et.al[15] present an weight estimation process with span time, request count and access sequence details. The user interest based page weight is used to extract the frequent item sets. Systolic tree is used to arrange candidate sets with frequency values. Due to the limited size of the systolic tree, a transactional database must be projected into smaller ones each of which can be mined in hardware efficiently. A high performance projection algorithm which fully utilizes the advantage of FP-growth is proposed and implemented. It reduces the mining time by partitioning the tree into dense and spare parts and sending the dense tree to the hardware. Systolic tree based rule mining scheme is enhanced for weighted rule mining process. Automatic weight estimation scheme is used in the system. With explosively growing number of Web contents including Digitalized manuals, emails pictures, multimedia, and Web services require a distinct and elaborate structural framework that can provide a navigational surrogate for clients as well as for servers. Due to the increasing amount of data Available online, the World Wide Web has becoming one of the most valuable resources for information retrievals and knowledge discoveries. So Sekhar Babu Boddu et.al [22] presents an introduction of Web mining as well as a review of the Web mining categories. Then we focus on one of these categories: the Web structure mining. Within this category, we introduce link mining and review two popular methods applied in Web structure mining: HITS and Page Rank.

## III. PROPOSED ARCHITECTURE

Proposed web page pre-fetching scheme use FP growth [2], weighted rule mining concept [] and Markov model [16] for fast and frequent web pre fetching .Proposed Scheme use FP growth tree concept for finding frequent page's efficiently without candidate set generation, whereas weighted mining concept are used to apply relative weight over each transaction after session and user identification .Markov Model use to assign that that relative weight over their relative position in transaction probability chain matrix suggested by markov model.

Proposed Scheme for web page prediction can be easily understand by the architectural diagram as show in figure 1. Where proposed diagram having two different layers, front and back. Front layer is use to grape web information ie web transaction information in web log. Whereas backed layer used to analyses this information and generate resultant Markov model for future web page prediction.

**Front Layer:-** Front layer responsible for capturing client web access behavior over web log file whereas back layer use this historical information as a input to analyses client web behavior for web page pre fetching. Once analysis over log file is completed, back layer generate resultant Markov model.

After completion of Markov model if any client A requests a web page $P_1$ web server performing two different job over that request before replying .first redirect that request to transaction probability matrix of backend layer, transaction probability matrix reply number of most frequent pre fetch page index number having higher relative weight where number of pre fetch page depend upon catch size ie as per requirement.

**Backend Layer:-** Back layer is use to pre- process and refine raw log file and generate Markov model .Back layer having following step

**Log Pre-Processing-** Log file used to capture client server behavior over the network at any time ie what page has been requested by which client and when all this information has been capture in web server log file. Along with that important information there is also some inconsistent data like noise, null value and other error information which is not so important for web personalization so in order improve web mining result its need to refine web log file before mining. Data cleaning, user and session identification, data integration and so on are main important part of log pre processing.

**Data Cleaning:-** Data cleaning is a process of removal of unwanted information ie not necessary for web pre-fetching like HTTP sound , picture and graphics information because page having sound .graphics and picture extension is not relevant for decision taking in web pre-fetching [6].

Data cleaning process also involves removal of unwanted failed HTTP status code. Status code is three digit code returns by server. Server serve status code in four different classes namely 200 Series , 300 series ,400 series and SOD

series where 200 series status code for successful transaction , 300 series code for redirect and 400 series code for failed authentication (401), Forbidden request for restricted subdirectory (403) and file not found (404,whereas SOD series code for server error.
In proposed methodology data cleaning process use to recognized useful token and remove unwanted and redundant token then store in data base after normalization.

**User & Session identification:-**
User and session identification is very important step towards web personalization generally IP address is used to distinguish but when there is an proxy server then number of user having same IP address then some more attribute like browses information ,operating system and Refer URI field is used as per concern[9].

**Weight Assignment** - Weight assignment concept is being used mapping any web page with their entire relevant page having higher relative weight.

Relative weight of any page y with respect to x means probability of page y request after page x is being calculated by dividing number of occurrence of page x with page y together with number of occurrence of page y.
**Transaction Probability Matrix**: - This step is concern with relative positioning of relative weight evaluate in previous step, whereas relative position is drive from Markov model. Actually data representation in Markov model is very efficient in both representation and retrieval.

The frequent patterns are extracted with the weight values. The weighted support is estimated and used for the pages. As suggested in Algorithm below.

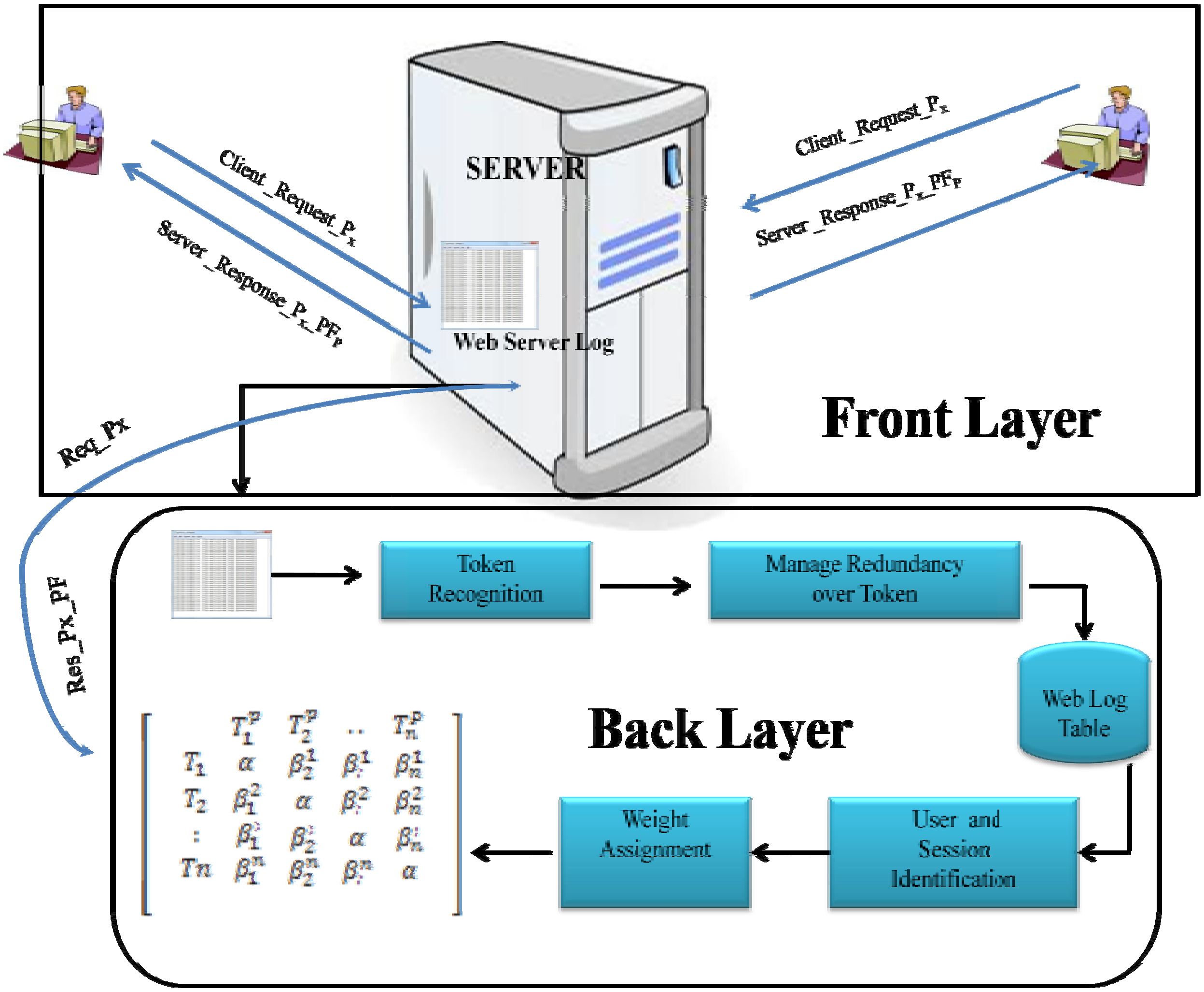

Figure: 1 Proposed Architecture

***Assumption***

*N=total number of line in log file*
*M= total number of attribute in log file*
*WL=web log file*
*T= Token in log file*
$R_{pi,pj}^{w}$ *= Relative weight Pj with respect to Pi*

$$P_{I,J}^{M} = \begin{bmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{bmatrix} = Matrix\ of\ markov\ model$$

```
// for inserting log value in database ie log pre processing
For (i=1;I<=n:i++)
{
For (j=1;j<=m:j++)
{

$T_j^i = \frac{dW_{L_i}}{dj} =$
$character\ stream\ between\ two\ separtor\ ( ; , / , \backslash , - , [ , ] )$
Insert $T_j^i$ $in\ web\ log\ table\ in\ databaseas\ Ith\ touple\ and$
Jth attribute
{
}
For (i=1;I<=n:i++)
{
For (j=1;j<=m:j++)
{
If ( $T_j^i$ contain style, graphics, and video, CSS, JS, picture
and sound file extension or contain status code above 200 )
{
Delete Ti record
Exit ()
}
}
}
// for weight assignment and session management
For (i=n;i<=n;i++)
{
$if\,(T_{IP}^{i}\ not\ in\ list(\ distinct\ User))$
$Add\ T^{i} in\ list\ (distinct\ user)$
$else\ if\,(T_{IP\ and\ os}^{i}\ not\ in\ list(\ distinct\ User))$
$Add\ T^{i} in\ list\ (distinct\ user)$
$else\ if\,(T_{IP\ ,os\ and\ browwser}^{i}\ not\ in\ list(\ distinct\ User))$
$Add\ T^{i} in\ list\ (distinct\ user)$
$else\ if\,(T_{IP\ ,os,browwser\ and\ referal\ uri}^{i}\ not\ in\ list(\ distinct\ User$
$))$
$Add\ T^{i} in\ list\ (distinct\ user)$
$Else$
$delete\ T^{i}\ from\ weblog\ table$
}
}
}
// for inserting value in matrix of Markov model
For (i=n;i<=n;i++)
{
For (j=n;j<=n;j++)
{
$R_{pi,pj}^{w} = \frac{total\ number\ of\ occurance\ of\ Pi\ \&\ Pj\ together}{total\ number\ of\ occurance\ of\ Pi}$
$P_{I,J}^{M} = \begin{bmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{bmatrix} = R_{pi,pj}^{w}$
}
}
```

## IV. SIMULATION AND RESULTS

For simulation and result analysis a real time scenario of client server architecture having 30 clients and one server is taken as a scenario for verification of proposed work whole verification is done over MATLAB 10 and used My Sql for data base support.

Proposed scheme use Markov model based virtual 2-D table that encapsulate relative weight of each page with each other.

**Time complexity –** Proposed methodology for taking decision about pre-fetch page having some extra overhead time required to evaluating the request from any client. In existing Systolic tree concept use an systolic Tree to store relative weight and time taken for taking decision about pre-fetch page is O (Log N) where N is height of tree. Whereas proposed technique used 2D table that take O (1) for pre-fetching single page same as Markov model. As per shows in figure 2 and Table 1.

Table 1: Time Comparison

| No. of page | Proposed Technique | Weighted Rule Model | Markov model |
|---|---|---|---|
| 1 | 1 | 0 | 1 |
| 10 | 1 | 1 | 1 |
| 20 | 1 | 1.301029996 | 1 |
| 50 | 1 | 1.698970004 | 1 |
| 100 | 1 | 2 | 1 |
| 150 | 1 | 2.176091259 | 1 |
| 200 | 1 | 2.301029996 | 1 |
| 250 | 1 | 2.397940009 | 1 |
| 300 | 1 | 2.477121255 | 1 |
| 350 | 1 | 2.544068044 | 1 |

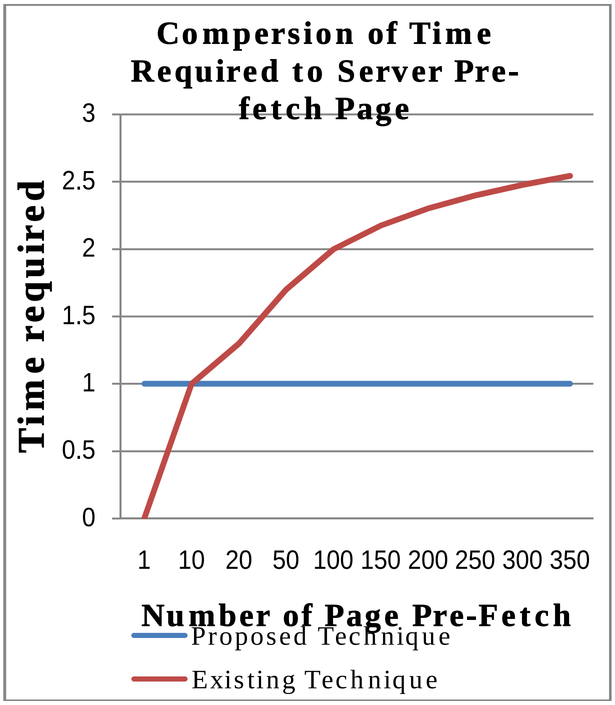

**Figure 2: Comparison of Time Required to Server Pre-fetch Page**

**Space Complexity-** In terms of space we need large space as compare to weighted tree concept but much lesser than plain Markov model so proposed methodology is moderate in space complexity. The graph shows the space complexity of the previous methods and the proposed method in figure 3. Here proposed method required much lesser space than markov model but little bit more than weighted tree Concept model.

Table2 : Space Comparison

| Number of pages | Proposed Technique | Weighted Rule | Markov model |
|---|---|---|---|
| 2 | 4 | 4 | 0.707106781 |
| 4 | 16 | 8 | 1.837117307 |
| 6 | 36 | 12 | 9.882117688 |
| 8 | 64 | 16 | 80.21178023 |
| 10 | 100 | 20 | 869.8739234 |

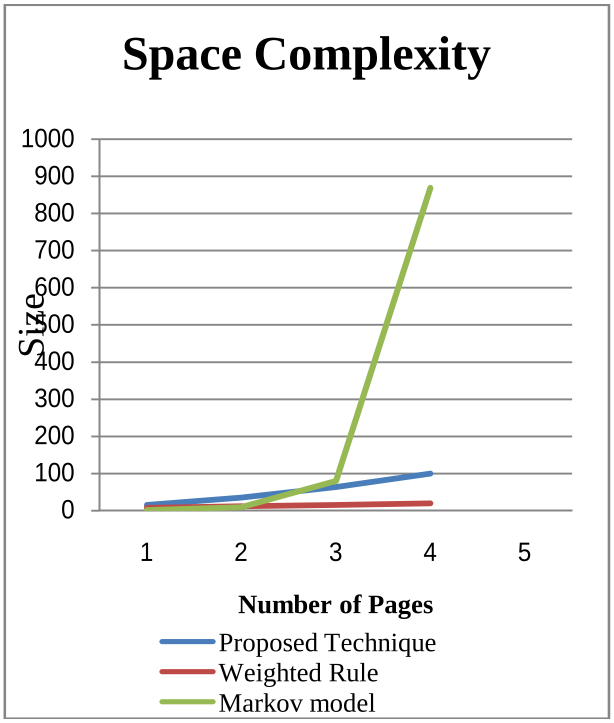

**Figure 3: Comparison of Space complexity for Pre-fetch Page**

## V. CONCLUSION

This paper proposed a method using FP growth Tree and Markov Model along with relative weight concept in order to apply the pre-fetching in the web environment. Tests that have been conducted in this proposed work using the Markov models shows that it gives better results as compare to previous work ie having moderate time and space complexity as compare to previous one. The implementation also shows that it is easy to apply in order to pre-fetch the page of a web site.

In future it is possible to apply some other methodology to generate the rules.

## VI. ACKNOWLEDGEMENT

I would like to say thanks to my guide “Prof.” who gives their knowledge and time in order to complete this paper. This paper will never complete without the support faculty member of CSE department of, Bhopal.

## REFERENCES

[1] R. Kosala and H. Blockheel, “Web Mining Research: A Survey”, In SIGKDD Explorations, Volume 2, Number 1, pages 1-15, 2000.

[2] P. Adriaans, D. Zantinge, “Data Mining” Addison Wesley Longman Limited, Edinbourgh Gate, Harlow, CM20 2JE, England. 1996.

[3] S. Chakrabarti, “Data mining for hypertext: A tutorial survey”. ACM SIGKDD Explorations, 1(2):1-11, 2000.

[4] M. Craven, D. Dipasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery, “Learning to extract symbolic knowledge from the World Wide Web”, In proceeding of the Fifteenth National Conference on Artificial Intelligence (AAAI98), pages 509-516, 1998.

[5] Pablo Rodriguez, Christian Spanner, and Ernst W. Biersack, "Analysis of Web Caching Architectures: Hierarchical and Distributed Caching", IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 9, NO. 4, AUGUST 2001

[6] L. Ramaswamy, A. lyengar, L. Liu, F. Douglis, “Automatic Fragment Detection in Dynamic Web Pages and Its Impact on Caching”, IEEE Transactions On Knowledge And Data Engineering, Vol. 17, No. 6, June 2005.

[7] P. Kolari and A. Joshi, “Web mining: Research and practice”, Computer Science Engineering .July/August (2004) 42–53

[8] B. Liu and K. Chang, “Editorial: Special issue on web content mining”, SIGKDD Explorations 6(2) 2004, pp 1–4.

[9] Nacim Fateh Chikhi, Bernard Rothenburger, Nathalie Aussenac-Gilles “A Comparison of Dimensionality Reduction Techniques for Web Structure Mining”, Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence 2007, pp 116-119.

[10] Lefteris Moussiades, Athena Vakali, "Mining the Community Structure of a Web Site," bci Fourth Balkan Conference in Informatics 2009, pp.239-244.

[11] Toufiq Hossain Kazi, Wenying Feng and Gongzhu Hu, “Web Object Prefetching: Approaches and a New Algorithm”, IEEE 2010, pp 115-120.

[12] Brijendra Singh and Hemant Kumar Singh, “Web Data Mining Research: A Survey”, IEEE 2010.

[13] Kavita Sharma, Gulshan Shrivastava and Vikas Kumar, “Web Mining: Today and Tomorrow”, IEEE 2011, pp 399-403.

[14] WANG Yong-gui and JIA Zhen, “Research on Semantic Web Mining” IEEE 2010, pp 67-70.

[15] P. Sampath, C. Ramesh, T. Kalaiyarasi, S. Sumaiya Banu and G. Arul Selvan, “An Efficient Weighted Rule Mining for Web Logs Using Systolic Tree”, IEEE 2012, pp 432-436.

[16] Nizar R. Mabroukeh and C. I. Ezeife, “Semantic-rich Markov Models for Web Prefetching”, IEEE 2009, pp 465-470.

[17] A.B.M.Rezbaul Islam and Tae-Sun Chung, “An Improved Frequent Pattern Tree Based Association Rule Mining Technique”, IEEE 2011.

[18] R.Agrawal, and R.Srikant, “Fast algorithms for mining association rules”, In VLDB’94, pp. 487-499, 1994 Borges and M. Levene,”A dynamic clustering-based markov model for web usage Mining”, cs.IR/0406032, 2004.

[19] Zhu, J., Hong, J. and Hughes, J. G. (2002a) Using Markov Chains for Link Prediction in Adaptive Web Sites. In Proc. of Soft-Ware 2002: the First International Conference on Computing in an Imperfect World, pp. 60-73, Lecture Notes in Computer Science, Springer, Belfast, April.

[20] K.Ramu, Dr.R.Sugumar and B.Shanmugasundaram “A Study on Web Prefetching Techniques” Journal of Advances in Computational Research: An International Journal Vol. 1 No. 1-2 (January-December, 2012)