A Cluster based Probabilistic Model for Link Prediction to Improve User Interface over Internet

Vivek Rawat
Research scholar
SIRT Bhopal

Sumit Vashishtha
Assistant professor
SIRT Bhopal

ABSTRACT

Rapid growth of web application has increased the researcher's interests in today's world. The world hasbeen surrounded by the computer's network. There exists a very useful application call web application that is used for the purpose of communication and data transfer. An application that is accessed with the help of web browser over a network is called as the web application. Web caching is considered to be the well-known strategy for improving the performance of Web based system. This performance is improved by keeping the Web objects that are likely to be used in the near future in location that is closer to user. The Web caching mechanisms therefore are implemented at three levels namely: (i) client level, (ii) proxy level and (iii) original server level. Significantly, proxy servers play the vital roles between users and web sites in reducing the response time of user requests as well as saving of network bandwidth. Thus, for achieving the better response time, an efficient caching approach must be implemented in a proxy server. This paper further includes weighted rule mining concept, cluster based link prediction and Markov model for fast and frequent web pre fetching.

Keywords

Web Services, Pre-fetching, Log file, cluster

1. INTRODUCTION

Web is a key resource in order to share the information along the world. It has large number of news, advertisements, global connectivity between people and lots of knowledge for the students. This massive use of Web or WWW makes it more important in the world of research. Researcher has the challenge to make the web applications more efficient. Many researchers work on it and give new idea in order to give the better results from the previous one. This dissertation is also puts its best foot forward in this era [1,12,13].

There is a huge need to improve the response time of server for web applications. Current Web has a massive repository due to increase its use suddenly. It has to focus on both the quality and quantity of web contents. Even, when the speed of Internet has improved with the reduced costs, the traffic is getting heavier to a large extend. The information which is enormous makes it difficult to find the relevant information quickly. This led to the effort to improve the speed, by reducing the latency that makes the web more relevant and more meaningfully connected.[2,8,9] .

The Cache prefetching plays an important role in order to enhance the response time and make the application well-organized. The web prefetching is a technique which is usedin order to preprocess the requests of the user, before they are actually demanded. Therefore, the time that the user must wait for the documents that is requested can be reduced by hiding the request latencies. Pre-fetching is the method for reducing Latencies. The user always expects an interactive response, better satisfaction and quality of output. There are various approaches and algorithms have been proposed for improving the web performance [3, 10, 11].

The proposed work will use to predict fourth coming link to improve the user experiences and expedites users visiting speed. Predictive Web pre-fetching or link prediction refers to the method of deducing the upcoming page accesses of a client based on its past experience. In this work we 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. The approach works on the basis is web mining with the combination of clustering approach.

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 about the proposed architecture of the presented work. After this the simulation result has discussed. Finally paper concludes in the section eight's.

2. PREVIOUS WORK

Research over web mining and web pre-fetching is going very fast in last decades. Toufiq Hossain Kazi et.al [4] 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 [6] proposed a new and improved FP tree with a table and a new algorithm for mining of the association rules. This algorithm further mines all possible frequent item set without generating the FP tree which is conditional in nature. It also provides the frequency of frequent items, which is used to estimate the desired association rules, Whereas P. Sampath et.al[5] 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 SekharBabuBoddu et.al [7] 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.

3. PROPOSED ARCHITECTURE

Proposed link prediction and web pre-fetching method need to use transaction weight rule mining concept, Markov model and cluster based link prediction method.Proposed method use transaction weight mining concept to evaluate comparative weight between any pair of web and apply cluster based link prediction method for evaluating probability of upcoming link then apply Markov Model to assign relative probabilities any pair of web pages to their relative position in transaction probability matrix suggested by Markov model.

3.1 Weight Assignment

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

$$\mathrm{R_w} = \frac{\text{Number of occurance of page x and page y together}}{\text{Number of occurance of page x}}$$

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.

3.2 Clustering-based Link Prediction

Based on the topological structure method and characteristics of most social networks

- G= (V, E) that indicates a social network, where V is the set of nodes and E is the set of links.
- e=(x, y) ϵ E represents an interaction between nodes x and y.
- Γ (x) denotes the set of neighbors of vertex x
- Cά , Cβ , … Cm. denote all cluster labels in G.
- XCβ denotes that node x belongs to Cβ
- Com x,y = Γ (x) U Γ (y)

Conditional probability that nodes x and y belong to the same cluster label Cβ as equation 1

$$\mathrm{p}\big(\mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta} \big| \mathrm{comm}_{\mathrm{x,y}}\big) = \frac{\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta}\big)\ \mathrm{p}(\mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta})}{\mathrm{p}(\ \mathrm{comm}_{\mathrm{x,y}})}$$

Where $\mathrm{p}(\mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta})$ is the probability that X belong to Cβ & Y belong to Cβ , $\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}}\big)$is the Probability that community X & Y have some same neighbor node,$\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta}\big)$ number of common neighbors with the same cluster label Cβ , where

$$\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta}\big) = \frac{\mathrm{comm}_{\mathrm{x,y}}^{\mathrm{in}}}{\mathrm{comm}_{\mathrm{x,y}}}$$

And $\mathrm{comm}_{\mathrm{x,y}}^{\mathrm{in}}$ is set of common neighbors belonging to the same cluster with nodes x and so

$$\mathrm{p}\big(\mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta} \big| \mathrm{comm}_{\mathrm{x,y}}\big)\ \alpha\ \mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta}\big)$$

ie

$\mathrm{p}\big(\mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta} \big| \mathrm{comm}_{\mathrm{x,y}}\big)$ is directly propoational to $\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta}\big)$.

Then Score measure for disconnected nodes pair(x, y) to link prediction

$$\mathrm{S_{x,y}} = \frac{\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta}\big)\ \mathrm{p}(\mathrm{x}^{\mathrm{c}_\beta}, \mathrm{y}^{\mathrm{c}_\beta})}{\mathrm{p}\big(\mathrm{comm}_{\mathrm{x,y}} \big| \mathrm{x}^{\mathrm{c}_\alpha}, \mathrm{y}^{\mathrm{c}_\beta}\big)\ \mathrm{p}(\mathrm{x}^{\mathrm{c}_\alpha}, \mathrm{y}^{\mathrm{c}_\beta})}$$

3.3 Transaction Probability Matrix

Transaction markov score matrix is use store relative score of link prediction evaluate in previous step at relative position in matrix, which help further in fast retrieving of score for taking decision in relevant page for web pre fetching .

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

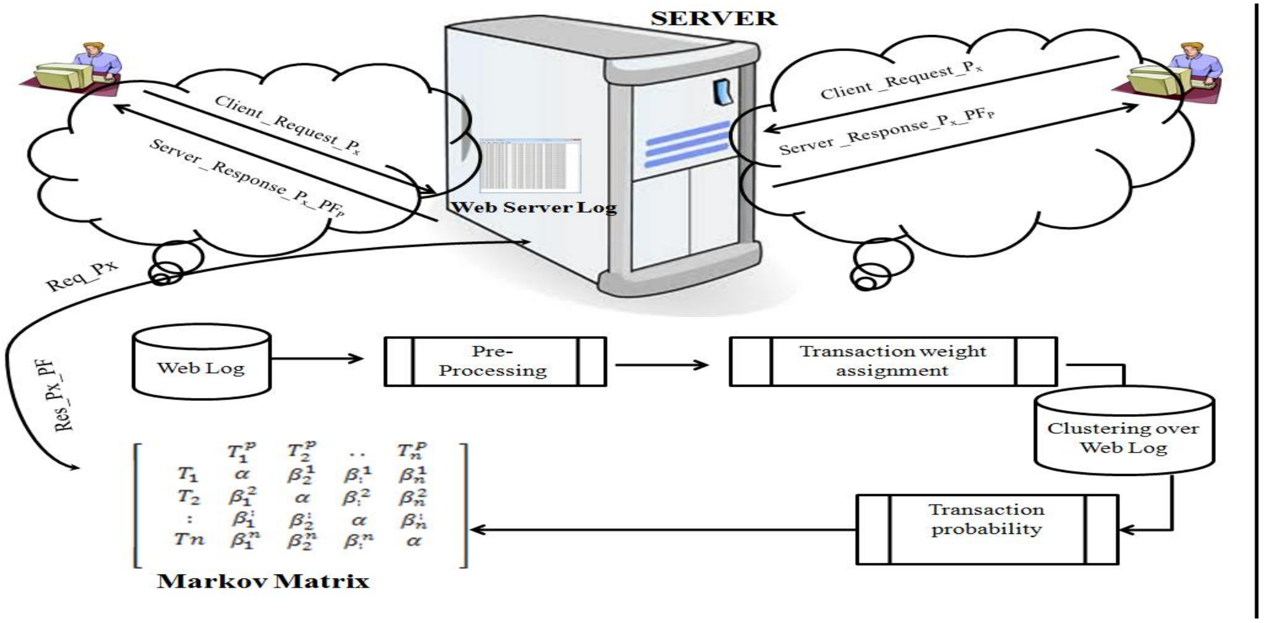

Figure: 1 Proposed Architecture

4. ALGORITHMS

Assumption

WL=web log file

C=cardinality of WL

P= degree of WL

$W^{R}_{Yi,Xj}$ = Relative weight Xj with respect to Yi

$S^{M}_{I,J} = \begin{bmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{bmatrix} = \text{Score Matrix}$

Algorithm

```
Step 1:-
For (i=1 to i<=C)// loop start for touple separation
{
For (j=1 to j<=P) // loop start for attribute separation
{
$C^{i}_{j}=\frac{dW_{L_i}}{dj}$ = logical entity between two separtor ( ; ,/,\
, −, [, ])
Insert   logical entity $C^{i}_{j}$ in web log table in databaseas   Ith
row and Jth column
}// loop end for attribute separation
} //loop end for touple separation
For (i=1 to i<=C) // loop start for touple extraction
{
For (j=1 to j<=P) // loop start for attribute extraction
{
Delete Ci record if its include style, graphics, and video, CSS,
JS, picture and sound file extension or contain status code
above 200
}// loop end for attribute extraction
} //loop end for touple extraction
Step 2:-
For (i=1 to i<=C)// loop start for session management
{
Case 1: − if $C^{i}_{IP}$ not in list( distinct User)
            Add $C^{i}$ in list (distinct user)
Case 2 : − $C^{i}_{IP\ and\ os}$ not in list( distinct User)
            Add $C^{i}$ in list (distinct user)
Case 3: − $C^{i}_{IP\ ,os\ and\ browwser}$ not in list( distinct User)
            Add $C^{i}$ in list (distinct user)
Case 4: − $C^{i}_{IP\ ,os,browwser\ and\ referal\ uri}$ not in list( distinct User
)
            Add $C^{i}$ in list (distinct user)
End case
                    delete $C^{i}$ from WL
}// loop end for session management
Step 3:-
For (i=1 to i<=C)// loop start for clustering and score
evaluation
{
Evaluate $p(xi^{c_\beta}, yi^{c_\beta})$
                    Evaluatep$(comm_{xi,yi})$
Evaluate
  $p(xi^{c_\beta}, yi^{c_\beta}|comm_{x,y}) = \frac{p(comm_{xi,yi}|xi^{c_\beta}, yi^{c_\beta})\ p(xi^{c_\beta}, yi^{c_\beta})}{p(\ comm_{xi,yi})}$
Evaluate
          $S_{xi,yi} = \frac{p(comm_{xi,yi}|xi^{c_\beta}, yi^{c_\beta})\ p(xi^{c_\beta}, yi^{c_\beta})}{p(comm_{xi,yi}|xi^{c_\alpha}, yi^{c_\beta})\ p(xi^{c_\alpha}, yi^{c_\beta})}$
}// loop end for clustering and score evaluation
Step 4:-
For (i=1 to i<=C) // loop for inserting value in score
probability matrix
{
For (j=1 to j<=P)
{
          $S_{xi,yi} = \frac{p(comm_{xi,yi}|xi^{c_\beta}, yi^{c_\beta})\ p(xi^{c_\beta}, yi^{c_\beta})}{p(comm_{xi,yi}|xi^{c_\alpha}, yi^{c_\beta})\ p(xi^{c_\alpha}, yi^{c_\beta})}$
$S^{M}_{I,J} = \begin{bmatrix} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \end{bmatrix} = S_{xi,yi}$
}
}
```

5. SIMULATION AND RESULTS

For simulation and result analysis we have considered a real time scenario of client server architecture that are 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 uses Markov model that is based on virtual 2-D table that encapsulate relative weight of each page with each other.

5.1 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

5.1.1 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 that 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.

Table 1: 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

This graph shows the time complexity of the various methods. It seems to us that the proposed method is more accurate than the other

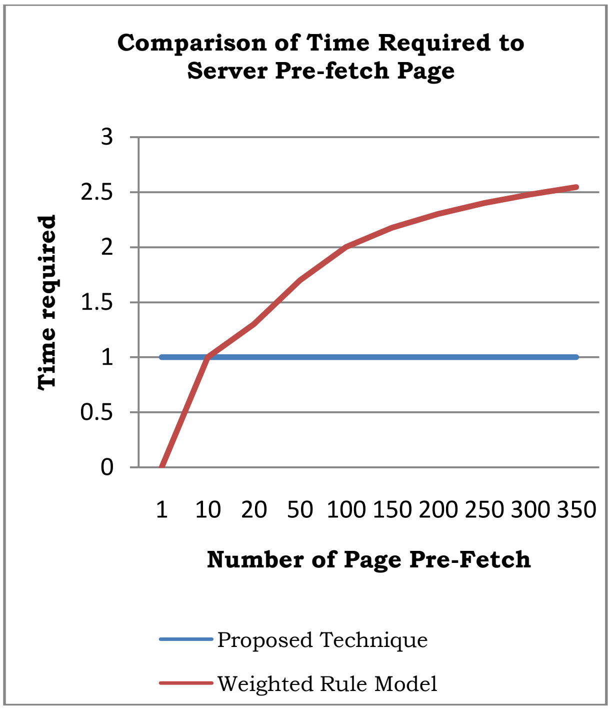

Figure 2 Graph for time complexity

In this graph we have compare the time complexity of various methods in order to prefetch the correct page. As shows in graph there are many techniques have done this work in previous time.

In terms of space we need large space so that it is not space efficient. This graph shows the space complexity of the previous methods and the proposed method. Here proposed method has high space complexity.

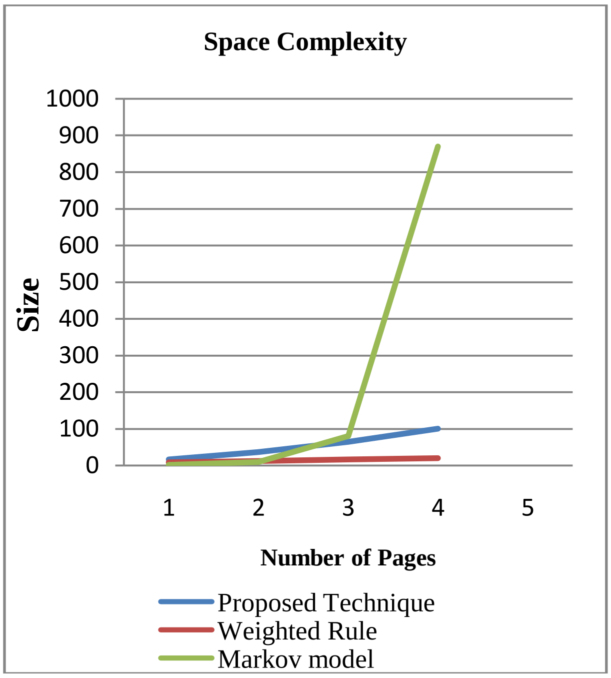

Figure 3:Graph to Show Space Complexity

We need large size of space due to use of huge data base. It also need large space for high number of rules used for pre-fetching the page.

6. CONCLUSION

The proposed method use weight based clustering approach for evaluating relative link prediction score between any two arbitrary link and use Markov Model concept for storing and retrieving score of link in order to predict upcoming web page. 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.

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

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

.

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

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

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

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

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

[11] 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.

[12] 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.

[13] 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)

[14]