© UNIVERSAL RESEARCH REPORTS | REFEREED | PEER REVIEWED
ISSN : 2348 - 5612 | Volume : 05 , Issue : 04 | January – March 2018

A Result Paper: Web Servers Optimization Using Web Prefetching

Ritu,
Student M.Tech(CSE)
J.I.E.T,Jind Haryana,India

Sapna Aggarwal
Assistant Professor J.I.E.T,
Jind Haryana, India

Abstract: ***The World Wide Web can be considered as a large distributed information system that provides access to shared data objects. As one of the most popular applications currently running on the Internet, The World Wide Web is of an exponential growth in size, which results in network congestion and server overloading.***

Keywords: Web prefetching, Page Rank, Web Server, Zipf Estimator.

1. Introduction

World Wide Web is an important area for data mining research due to the huge amount of information. The success of WWW depends on the response time. Due to the fast development of internet services and a huge amount of network traffic, it is becoming an essential issue to reduce World Wide Web user-perceived latency. Caching popular objects close to the users provides an opportunity to combat this latency by allowing users to fetch data from a nearby cache rather than from a distant server. Web caching has been recognized as one of the effective schemes to alleviate the service bottleneck and reduce the network traffic, thereby minimize the user access latency. Although web performance is improved by caching, the benefit of caches is limited. To further reduce the retrieval latency, web prefetching becomes an attractive solution to this problem. Prefetching reduces user access time, but at the same time, it requires more bandwidth and increases traffic. Zipf's law governs many features of the Internet. Observations of Zipf distributions, while interesting in and of themselves, have strong implications for the design and function of the Internet. The connectivity of Internet routers influences the robustness of the network while the distribution in the number of email contacts affects the spread of email viruses. Even web caching strategies are formulated to account for a Zipf distribution in the number of requests for Web pages.

This Paper implements a Zipf law based novel approach for the determination of next page likely to be accessed by specific client.

2. Motivation

In this work the main focus is put on the web usage mining technique (a technique of web mining), which is applied on the proxy server log to generate the preprocessed log and the users navigation patterns.

3. Litearture Review

[1]. However, several recent studies have investigated whether the requests do indeed follow Zipf's law and concluded otherwise [16], [2]. Padmanabhan [3] use dependency graph for prediction and prefetching. Their prediction algorithm construct a dependency graph that depicts the pattern of accesses to different file stored at the server. The graph has a node for every file that has ever been accessed. There is an arc from node A to B if and only if B was accessed with in w(look ahead window size) access after A. It was Etzioni [4] who first coined the term web mining .Etzioni starts by making a hypothesis that the information on the web is sufficiently structured and outlines the subtask of web mining. His paper describes the web mining process.S. Jespersen [6] said that Markov assumptions are used as the basis to mine the structure of browsing patterns. Markov-based structures for web usage mining are best suited for tasks demanding less accuracy such as pre-fetching, personalization, and targeted ads. Many of the papers proposed using association rules or Markov models for next page prediction. Faten Khalil [7] proposes an improved approach, based on a combination of Markov models and association rules that result in better prediction accuracy and more coverage. They used low order Markov models to predict multiple pages to be visited by a user and then applied association rules to predict the next page to be accessed by the user based on long history data.

4. Web Mining

Overview

The web is a vast collection of completely uncontrolled heterogeneous documents. Thus, it is huge, diverse, and dynamic, and raises the issues of scalability, heterogeneity, and dynamism, respectively. Due to these characteristics, we are currently drowning in information, but starving for knowledge; thereby making the web a fertile area of data mining research with the huge amount of information available online. Data mining refers to the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.

© UNIVERSAL RESEARCH REPORTS | REFEREED | PEER REVIEWED
ISSN : 2348 - 5612 | Volume : 05 , Issue : 04 | January – March 2018

5. Determing Next Page Access By Zipf Estimator

5.1 Introduction

The rapid increase of World Wide Web users and the development of services with high bandwidth requirements have caused the substantial increase of response times for users on the Internet. A resource retrieval on the World Wide Web (WWW) starts with a request issued by a client. The WWW server replies with any requested resources. The client parses this response and displays it. This interaction often takes a long time, which is called the WWW latency. Large WWW latency may cause user discontent.

WWW latency depends on network latency and performance of both servers and clients. Network latency depends on network bandwidth and propagation delay. Network bandwidth and host performance can be improved by upgrades. As the costs of network connectivity and computer performance have been decreasing dramatically, it will be easier to get broad-bandwidth networks and high-performance computers in the near future. But the propagation delay cannot be eliminated because the speed of light is constant. One solution is caching, a popular technology of WWW proxy servers. A cache stores resources that have been accessed recently. Therefore, caching improves retrieval time of frequently accessed resources. But its effect on WWW latency is small because frequently retrieved resources account for little retrieval. Several papers reported that the hit-rate of the caching proxy server is 30%-50%

Prefetching of the web pages is another potential research area that can reduce the web access latency. It refers to the process of deducing client future requests for web objects and getting those objects into the cache before it is explicitly requested. The major advantage of prefetching is that it prevents bandwidth under utilization. However without a carefully designed prefetching scheme, it may happen that several already transferred web pages might never be requested by the client. This would result in bandwidth wastage. In this paper, a prediction engine called Prediction Prefetching Engine (PPE) processes the past references to deduce the probability of future access for the documents accessed so far. In fact, it resides on the proxy server

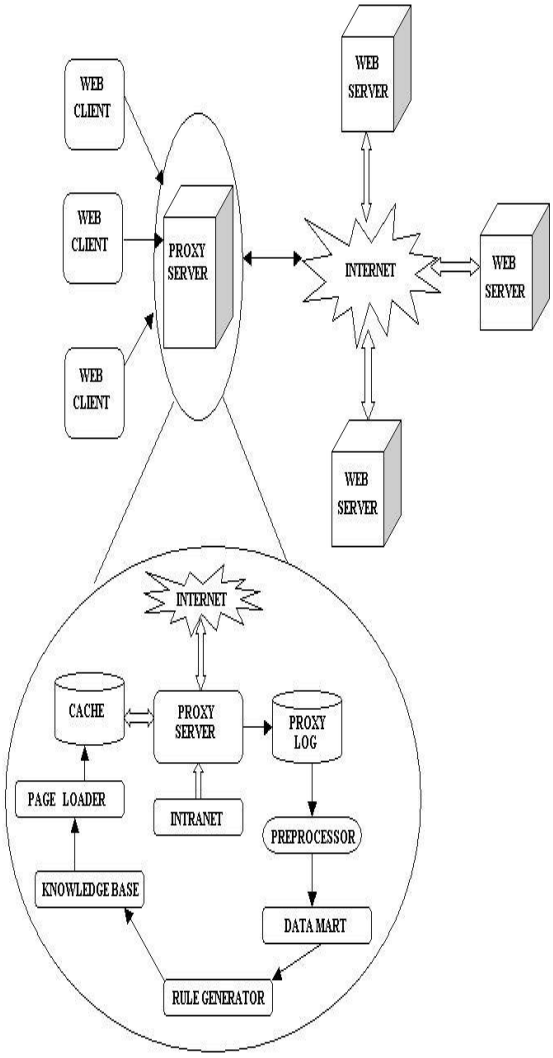

Fig 1 Prefetching the document at Proxy side

© UNIVERSAL RESEARCH REPORTS | REFEREED | PEER REVIEWED
ISSN : 2348 - 5612 | Volume : 05 , Issue : 04 | January – March 2018

5.2 Algorithms

Let P_i is the log retrieved from the proxy server. Now this unfiltered log is processed by applying the data preprocessing technique of web usage mining.

Algorithm 5.1 Transaction Preprocessor
Input: P_i
Output: $T_{j,,}$
Algorithm Preprocessor(P_i)

```
{
        index=0; user=0;session=0;
        for move =1 to n do
        {
                If there exist *.jpeg and *.gif
                file Continue;
                else
                 {
                Tj(index) = Pi (move);
                        Index++;
                }
           }
         for k=1 to index do
        {
                If Tj (k).ip== Tj(K+1).ip
                {
                User is same;
                continue;
                }
                else
                User is user++;
        }
      for m=1 to index do
        {
                If Tj (m).time <=30
                {
                Session is same;
                continue;
                }
                else
                Session  is  session++;
        }
        }
```

Let us assume that after transaction preprocessing steps like cleaning the web log, user and session identification the preprocessed log_i is formed and assume target set T_i. Let $Equivalance\_set_i$, $Lower\_set_i$, $Upper\_set_i$ are the equivalence set, lower approximation set, and upper approximation set with i number of moves

Algorithm 5.2 Clustering
Input*: Log[i], T[i]*
Output*:Equivalance_set[i][j], Lower_set[i] , Upper_set[i]*
Algorithm clustered_set*(Log[i], T[i], Equivalance_set[i][j], Lower_set[i], Upper_set[i])*

```
{
equivalent_set(Log[i], Equivalance_set[i][j], k );
lower_app(T[i] , Equivalance_set[i][j], k, Lower_set[i]);
upper_app(T[i] , Equivalance_set[i][j], k , Upper_set[i]);
}
```

Algorithm equivalent_set*(log[i], Equivalance_set[i][j] ,k)*

```
{
for i=1 to n-1 do{
   flag=false;
   for j=i+1 to n do{
      if (Log[i]==Log[j])
```

```
        flag=true;
      else {
        flag=false
        break;
      }
      if(flag==true){
        Equivalance_set[k][0]=i;
        Equvalancei_set[k][1]=j;
        k++;
      }
   }
}
}
```

Algorithm lower_app*(T[i] , Equivalance_set[i][j], k, Lower_set[i])*

```
{
for i=1 to k do{
   for j=0 to n do{
      if ((Equivalance_set[i][0]==T[j]) &&(Equivalance_set[i][1]==T[j]))
      {
        L[i][0]= Equivalance_set[i][0];
        L[i][1]= Equivalancei_set[i][1];
      }
   }
}
}
```

Algorithm upper_app*(T[i] , Equivalance_set[i][j], k, Upper_set[i])*

```
{
for i=1 to k do{
   for j=0 to n do{
        if ((Equivalance_set[i][0]==T[j]) || (Equvalancei_set[i][1]==T[j]))
        {
          Upper_set[i][0]= Equivalance_set[i][0];
          Upper_set[i][1]= Equvalancei_set[i][1];
        }
   }
}
}
```

Let us assume that after applying clustering the clustered sets $Upper\_set_i$ are formed .$Rules_i$ is the set of rules formed after applying markov and association.

Algorithm 5.3 Markov_&_association

Input*: Upper_set[i]*

Output*: Rules[i]*

Algorithm markov_association*(Upper_set[i])*

```
{
Build a low order Markov model
for each state of the Markov model
{
if the prediction is ambiguous
{
Collect all sessions satisfying the state Construct
association rules to resolve ambiguity Store the
association rules with the state
}

}
}
```

Let m_{i+1} is the next move, $Rules_i$ and S_i are transaction set and statistic table with i number of moves respectively and $Rank(m_{i+1})$ is the rank of m_{i+1}.

Algorithm 5.4 Zipf_estimator

Input: m_{i+1} , $Rules_i$
Output: S_{i+1}
Algorithm Zipf_estimator (m_{i+1}, $Rules_i$, S_i)
{
count (m_{i+1}) = 0
for i=1 to n do
{
If there exist m_{i+1} in $Rules_i$
{
count(m_{i+1}) =count(m_{i+1})+1
freq_count := count (m_{i+1}) / i+1
if (freq_count(m_{i+1}) = = freq_count (S_k)

{
Rank (m_{i+1}) := Rank(S_k)
}
else
{
Rank (m_{i+1}):= Rank (S_k) + 1 , where k= 1<=k<=i
}
S_i = < S_1,..., S_k ,....,S_i >
}
else
{
freq_count := 1/ i+1;
if (freq_count(m_{i+1}) = = freq_count (S_{i-1})
{
Rank (m_{i+1}):= Rank (S_{i-1})
}
else
{
Rank (m_{i+1}):= Rank (S_i) + 1
}

S_i = < S_1.S_2, ...,S_i , S_{i+1}>
}
P (m_{i+1}) = Rank(m_{i+1}) +
}
}

After determining the probability of the page the page with the highest probability can be prefetched in the cache of the proxy server

5.3 Implementation Details

An architecture developed in the implementation stage suggested the prefetching the web pages from WWW on the proxy server with the Zipf estimator can make the pronounceable change in the predicting and prefetching the web pages from the proxy server. However the test collection was too small to allow the effectiveness of the Zipf estimator to be assessed.

Furthermore it covers:

- Demonstrate the use of Zipf estimator to calculate the probability of web page that is to be prefetched by the proxy server.
- To evaluate the effectiveness of the Zipf estimator

We have divided the system in to two parts: Rule formulator and Rule Selector.

Rule Formulator: By applying the data mining techniques such as clustering, Markov model and association rule the rule is formed from the data mart. Data mart is the cleaner version of the proxy log after preprocessing.

Rule Selector: The rules formed by the rule formulator are extracted by the rule selector. Rule selector phase is further divided in to two phase: Rank analysis and the probability calculator. For calculating the probability the Zipf estimator is implemented.

Zipf estimator is based on Zipf law. Zipf's Law states that frequency of terms in a set of text collection follows a power law distribution. By the Zipf estimator the probability of accessing the next page can be computed efficiently.

© UNIVERSAL RESEARCH REPORTS | REFEREED | PEER REVIEWED
ISSN : 2348 - 5612 | Volume : 05 , Issue : 04 | January – March 2018

5.4 RESULTS

Using markov model we can determine that there is a 50-50 chance that the next page to be accessed by the user accessing the page course and counseling could be either admission or curriculum. Whereas association rule take this result a step further by determining that if user access page facility before course and counseling then there is a100% confidence that the user will access admission next. Whereas if the user visit page faculty before visiting page course and counseling, then there is a 100% confidence that user will access page curriculum.html next. In the same way other rule can be formulated.

Table 1: Statistic table

	moves	Freqcount(F)	Rank(R)	Probability (P)
R1	home-> facility	0.61	1	0.06
R2	academic- >home	0.46	2	0.03
R3	facility- >course	0.23	3	0.02
R4	home-	0.15	4	0.015
	>course			
R5	counseling- >admission	0.07	5	0.011
R6	admission- >placement	0.07	5	**?** 0.011

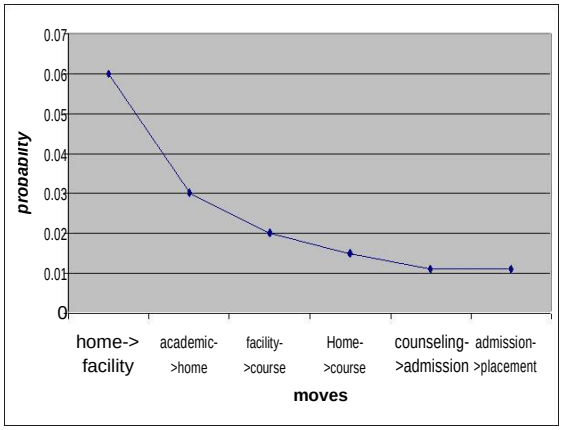

Fig 2: Zipf Curve

6. Conclusion

This Paper views the web as a system for the prediction and prefetching the useful information which is helpful for the user. It emphasis on the zipf estimator (a probability calculator), for estimating the probability of accessing the next page. Depending upon the probability of the next page, the page can be prefetched locally on the proxy server. When the user request for that page the page is given directly to the user rather than going to the web server.

Future Work

Usage patterns discovered through Web usage mining are effective in capturing item-to-item and user-to-user relationships and similarities at the level of user sessions. However, without the benefit of deeper domain knowledge, such patterns provide little insight into the underlying reasons for which such items or users are grouped together. Furthermore, the inherent and increasing heterogeneity of the Web has required Web-based applications to more effectively integrate a variety of types of data across multiple channels and from different sources.

Thus, a focus on techniques and architectures for more effective integration and mining of content, usage, and structure data from different sources is likely to lead to the next generation of more useful and more intelligent applications, and more sophisticated tools for Web usage mining that can derive intelligence from user transactions on the Web.

References

[1] Virgilio Almeida, Azer Bestavros, Mark Crovella, and Adriana de Oliveira, " Characterizing reference locality in the WWW " , In IEEE International Conference in Parallel and Distributed Information Systems, Miami Beach,

Florida, USA, December 1996. http://www.cs.bu.edu/groups/oceans/papers/ Home.html

[2] Virgilio Augusto F. Almeida, Marcio Anthony G. Cesirio, Rodrigo Fonseca Canado, Wagner Meira Junior, and Cristina Duarte Murta, "Analyzing the behavior of a proxy server in the light of regional and cultural issues." http://www.anades.dcc.ufmg.br/paperSubmetidos/ web cache/cultural/, 1998.

[3] Padmanbhan, "Using Predictive prefetching to Improve World wide web Latency", V.N, 1996 Comput. Comm. Rev, 26(3):22-36

[4] O. Etzioni, "The World Wide Web: Quagmire or gold mine". Communication of the ACM, 39(11): 65-68, 1996.

[5] Carlos Cunha, Azer Bestavros, and Mark Crovella, "Characteristics of WWW client-based traces.", Technical Report TR-95-010, Boston University, Computer Science Dept., Boston, MA 02215, USA, April 1995.

[6] S. Jespersen, T. B. Pedersen, and J. Thorhauge, "Evaluating the markov assumption for web usage mining," in WIDM '03: Proceedings of the 5th ACM international workshop on Web information and data management. New York, NY, USA: ACM Press, 2003, pp. 82-89

[7] Faten Khalil, Jiuyong Li and Hua Wang "A Framework of Combining Markov Model with Association Rules for Predicting Web Page Accesses" ,Proc. Fifth Australasian Data Mining Conference (AusDM2006), CRPIT Volume 61,177-184

[8] Pei Cao,Edward W. Felten,Anna R. Karlin, Kai Li, "A study of integrated prefetching and caching strategies" Proceedings of the 1995 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems. Pages: 188 - 197, 1995

[9] Steven Glassman, " A caching relay for the World Wide Web", In First International Conference on the World Wide Web, CERN, Geneva, Switzerland, May 1994.