Research Paper on Web Mining using Zipf Estimator

Ankita Gulati[1], Ms. Sarul[2]

[1]M. Tech. Student, Department of CSE, RIEM, ROHTAK, Haryana
[2]Assistant Professor, Department of CSE, RIEM, ROHTAK, Haryana

I. INTRODUCTION

The wide adoption of Internet has fundamentally altered the ways in which we communicate, gather information, conduct businesses and make purchases. As the use of the World Wide Web, computer scientists and physicists rushed to characterize this new phenomenon. While initially they were surprised by the tremendous variety the Internet demonstrated in the size of its features, they discovered a widespread pattern in their measurements.

The expansion of the World Wide Web (Web for short) [1] has resulted in a large amount of data that is now in general freely available for user access. The different types of data have to be managed and organized in such a way that they can be accessed by different users efficiently. Therefore, the application of data mining techniques on the Web is now the focus of an increasing number of researchers. Several data mining methods are used to discover the hidden information in the Web. However, Web mining does not only mean applying data mining techniques to the data stored in the Web. The algorithms have to be modified such that they better suit the demands of the Web. New approaches should be used which better fit the properties of Web data [2]. Furthermore, not only data mining algorithms, but also artificial intelligence, information retrieval and natural language processing techniques can be used efficiently.

II. LITRATURE REVIEW

It was Etzioni [3] 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. Cooley et al; Srivastava et al. [4] define Web usage mining as a three-phase process, consisting of preprocessing, pattern discovery, and pattern analysis. Their prototype system, Web SIFT, first performs intelligent cleansing and preprocessing for identifying users, server sessions, and inferring cached page references through the use of the referrer field, and also performs content and structure preprocessing [5]. Wang *et al.* in [6] surveys the caching studies taking into account many issues such as caching architectures, replacement policies, cache routing, dynamic caching, fault tolerance, security, etc. Padmanabhan [2] 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. M. Eirinaki [7] use Association rules are used in order to discover the pages which are visited together even if they are not directly connected, which can reveal associations between groups of users with specific interest. MOBASHER [8] showed the evidence that web request follow a Zipf-like distribution. He first investigates the page request distribution seen by web proxy cache using traces from a variety of sources. Lei Shi [9] presents the Web object popularity based model on zipf –like law, introduces the stability concept of the web system, and calculate the upper bound for the minimum length of the request stream in order to get stability.

III. PROPOSED SCHEME

For finding the page to be prefetched first the proxy server raw log is processed using the data preprocessing technique of web usage mining. On the preprocessed log various web mining techniques such as rough set theory for finding clusters, markov and association rule mining is applied on the clustered set of the pages. Applying all these techniques [10-13] the frequently visited pattern are determined. On these patterns the zipf estimator is applied to determine the probability of next page access and that pages are prefetched in the cache.

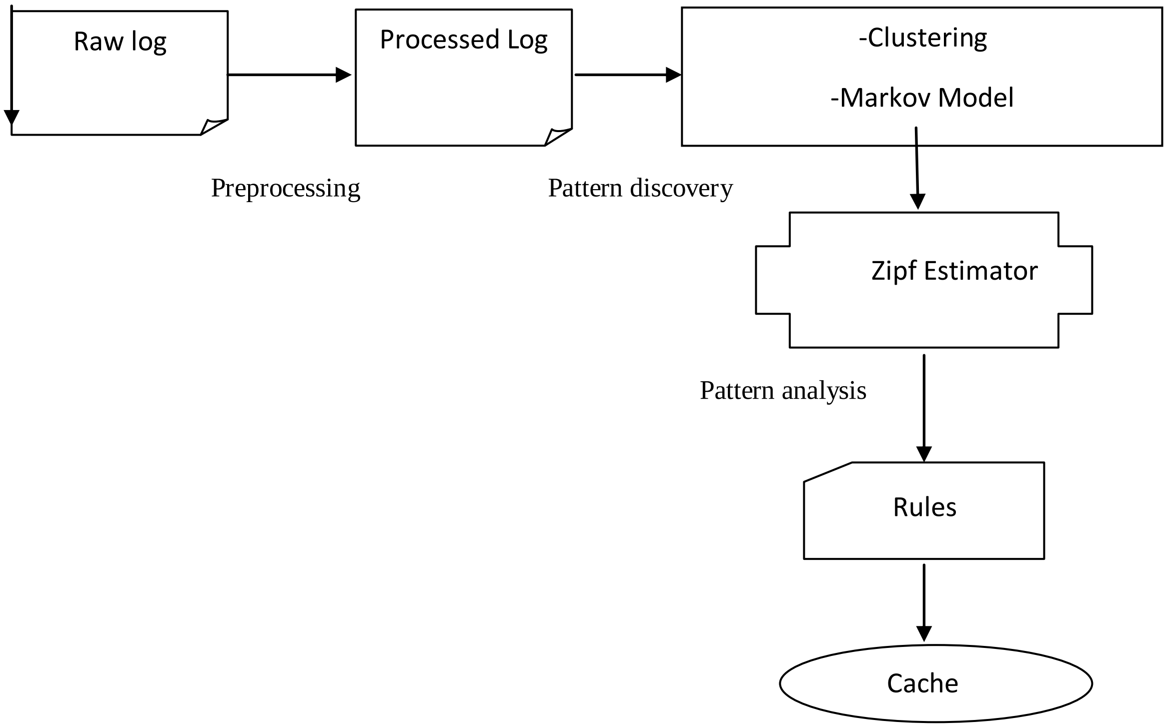

Figure 1.1: Work Flow of implementation

The basic operating principle of a proxy server is quite simple: It is server which acts as a "proxy" for an application by making a request on the Internet in its stead. This way, whenever a user connects to the Internet using a client application configure to use a proxy server, the application will first connect to the proxy server and give it its request. The proxy server then connects to the server which the client application wants to connect to and sends that server the request. Next, the server gives its reply to the proxy, which then finally sends it to the application client.

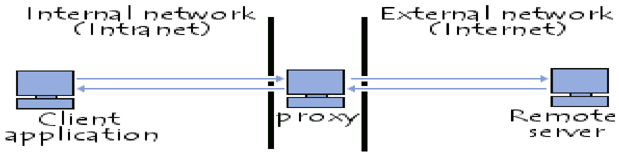

Figure 1.2: Operating Principle of Proxy Server

WEB MINING:

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.

Web mining can be broadly defined as the discovery and analysis of useful information from the WWW. In web mining data can be collected at the server side, client side, proxy servers, or obtained from an organization's database. Depending on the location of the source, the type of collected data differs [33].It also has extreme variation both in its content (e.g., text, image, audio, symbolic) and meta information that might be available. This makes the techniques to be used for a particular task in web mining widely varying. Some of the characteristics of web data are : 1) Unlabeled; 2) Distributed; 3) Heterogeneous (mixed media); 4) Semi structured; 5) Time varying; 6) High dimensional

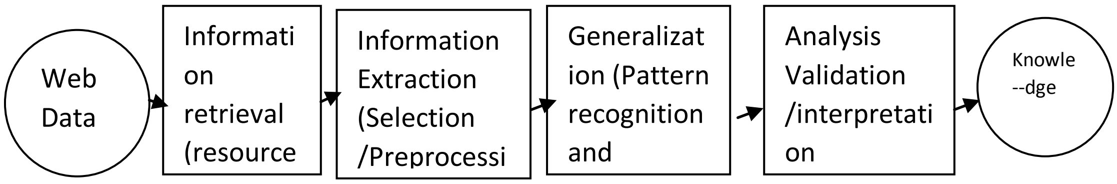
Web Data
Information retrieval (resource
Information Extraction (Selection /Preprocessi
Generalization (Pattern recognition and
Analysis Validation /interpretation
Knowle --dge

Figure 1.3: Web mining subtask

1) **Information Retrieval (IR) (Resource Discovery):** Resource discovery or IR deals with automatic retrieval of all relevant documents, while at the same time ensuring that the non-relevant ones are fetched as few as possible. The IR process mainly includes document representation, indexing, and searching for documents. An index is, basically, a collection of terms with pointers to places where the information about documents can be found.
2) **Information Selection/Extraction and Preprocessing:** Once the documents have been retrieved the challenge is to automatically extract knowledge and other required information without human interaction. Information extraction (IE) is the task of identifying specific fragments of a single document that constitute its core semantic content.
3) **Generalization:** In this phase, pattern recognition and machine learning techniques are usually used on the extracted information. Most of the machine learning systems, deployed on the web, learn more about the user's interest than the web itself. A major obstacle when learning about the web is the labeling problem: data is abundant on the web but it is unlabeled. Many data mining techniques require inputs labeled as positive (yes) or negative (no) examples with respect to some concept.
4) **Analysis:** Analysis is a data-driven problem which presumes that there is sufficient data available so that potentially useful information can be extracted and analyzed. Humans play an important role in the information or knowledge discovery process on the web since the web is an interactive medium. This is especially important for validation and or interpretation of the mined patterns which take place in this phase.

Web mining can be categorized in to three area of interest based on which part of the web to mine: a) **Web content mining ;** b) **Web usage mining ;** c) **Web structure mining**

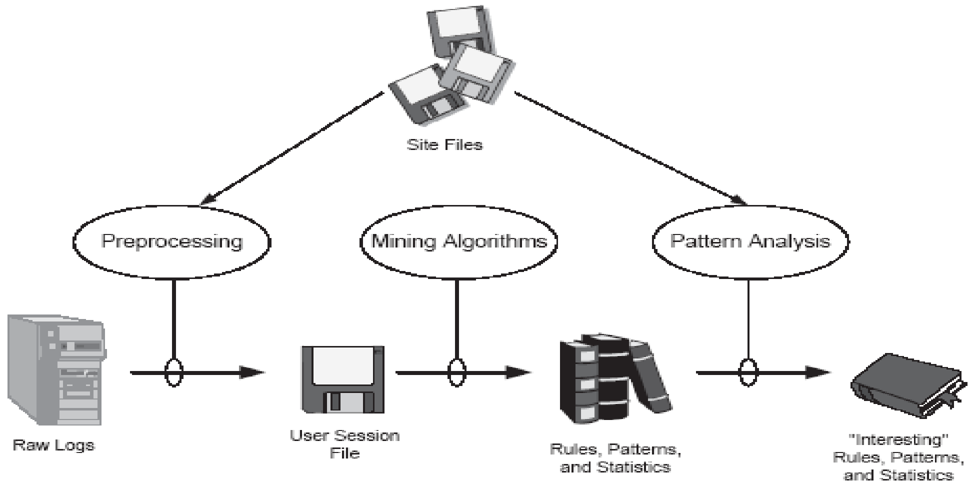

Figure 1.4: Web Usage Mining

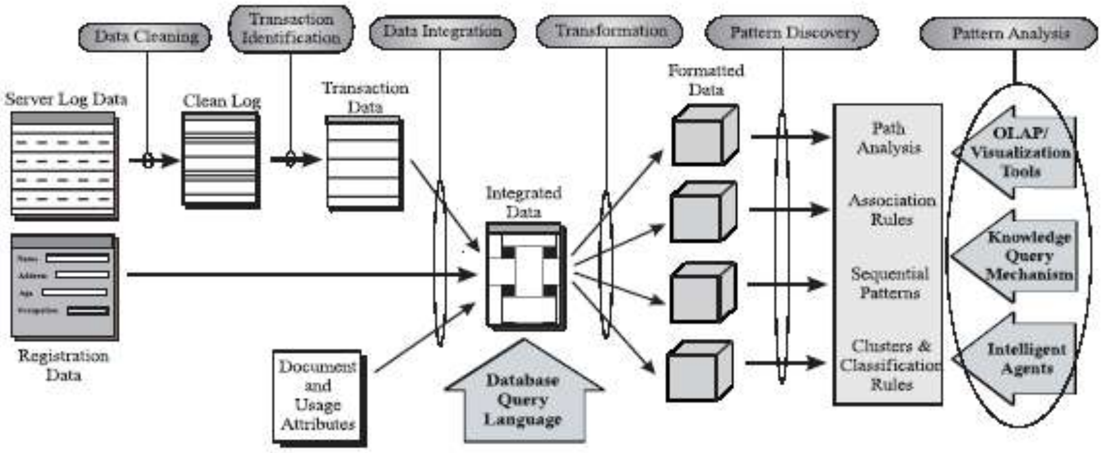

Figure 1.5: Data Preprocessing

Proxy server log: The records of all the users/clients that send requests to the server are kept on the web logs which are used to form the mineable warehouse. These warehouses are used to track the user activity. The user activity is tracked and recorded in these warehouses in the form of logs maintained by the proxy server. Since a proxy server sits between the client and the web server, it is comparatively easy to manage than the web server logs especially when it has to maintain the log for limited clients.

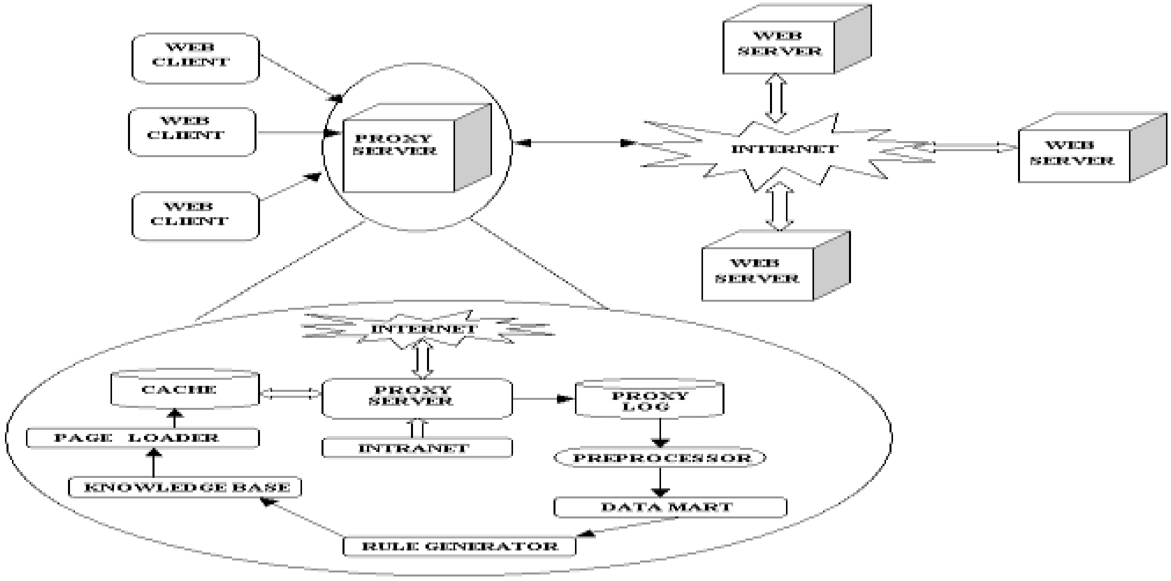

Figure 1.6: Prefetching the document at Proxy side

.

IV. METHODOLOGY

K-Order Markov Model and Association Rule: After dividing user sessions into a number of clusters, Markov model analyses [14-15] 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. Let $P = \{p1, p2, \ldots\ldots\ldots, pm\}$ 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 $(p_i \mid W)$ is the probability that the user visits pages p_i next. Page P_{l+1} the user will visit next is estimated by:

$$P_{l+1} = \text{argmax}_{p \in IP}\{P(P_{l+1}=p|W)\}$$

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

This probability, prob $(p_i \mid W)$ 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 $(p_i \mid W)$. 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 pi 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 become

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

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 *kth* 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.

Let S^k_j be a state containing k pages, $S^k_j = (p_{l-(k-1)},p_{l-(k-2\_},\ldots.p_l\ )$ The probability of $P(p_i \mid S^k_j)$ is estimated as follows from a history (training) data set.

$$P(p_i \mid S^k_j) = \frac{\text{Frequency}(<S^k_{j,,}\ p_i>)}{\text{Frequency}(S^k_j)} \quad (3)$$

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.

V. IMPLEMENTATION AND RESULT

An architecture developed in the implementation stage suggested the prefetching the web pages from WWW on the proxy server with the Zipf estimator [16-18] 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

OUR APPROACH: 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.

RESULTS

A Raw proxy log is show in the table below:

Table 1.1: Raw Proxy Log

I P Address Userid Time Method/ URL/ Protocol Status Size Referred Agent
1 123.456.78.9 - [25/Apr/2009:03:04:41 -0500] "GET home.html HTTP/1.0" 200 3290 - Mozilla/3.04 (Win95, I)
2 123.456.78.9 - [25/Apr/2009:03:05:34 -0500] "GET carrier.html HTTP/1.0" 200 2050 home.html Mozilla/3.04 (Win95, I)
3 123.456.78.9 - [25/Apr/2009:03:05:39 -0500] "GET admission.html HTTP/1.0" 200 4130 - Mozilla/3.04 (Win95, I)
4 123.456.78.9 - [25/Apr/2009:03:05:40 -0500] "GET inst.jpg HTTP/1.0" 200 4130 - Mozilla/3.04 (Win95, I)
5123.456.78.9 - [25/Apr/2009:03:06:41 -0500] "GET image.gif HTTP/1.0" 200 4130 - Mozilla/3.04 (Win95, I)
6 123.456.78.9 - [25/Apr/2009:03:06:02 -0500] "GET faciltiy.html HTTP/1.0" 200 5096 B.html Mozilla/3.04 (Win95, I)
7 123.456.78.9 - [25/Apr/2009:03:06:58 -0500] "GET home.html HTTP/1.0" 200 3290 - Mozilla/3.01 (X11, I, IRIX6.2, IP22)

8123.456.78.9 - [25/Apr/2009:03:07:42 -0500] "GET carrier.html HTTP/1.0" 200 2050 home.html Mozilla/3.01 (X11, I, IRIX6.2, IP22)
9123.456.78.10 - [25/Apr/2009:03:07:55 -0500] "GET cse.html.html HTTP/1.0" 200 8140 admission.html Mozilla/3.04 (Win95, I)
10 123.456.78.10 - [25/Apr/2009:03:09:50 -0500] "GET placement.html HTTP/1.0" 200 1820 home.html Mozilla/3.01 (X11, I, IRIX6.2, IP22)
11 123.456.78.11 - [25/Apr/2009:03:10:02 -0500] "GET hostel.html HTTP/1.0" 200 2270 facility.html Mozilla/3.04 (Win95, I)
12 123.456.78.11 - [25/Apr/2009:03:10:45 -0500] "GET course.html HTTP/1.0" 200 9430 placement.html Mozilla/3.01 (X11, I, IRIX6.2, IP22)
13 123.456.78.11 - [25/Apr/2009:03:12:23 -0500] "GET counselling.html HTTP/1.0" 200 7220 Carrier.html Mozilla/3.04 (Win95, I)
14 123.456.78.11 - [25/Apr/2009:05:05:22 -0500] "GET home.html HTTP/1.0" 200 3290 - Mozilla/3.04 (Win95, I)
15 123.456.78.11 - [25/Apr/2009:05:06:03 -0500] "GET examination.html HTTP/1.0" 200 1680 home.html Mozilla/3.04 (Win95, I)

After cleaning the unwanted file we get the reduced log.

Table 1.2: Table after data cleaning

I P Address Userid Time Method/ URL/ Protocol Status Size Referred Agent
1 123.456.78.9 - [25/Apr/2009:03:04:41 -0500] "GET home.html HTTP/1.0" 200 3290 - Mozilla/3.04 (Win95, I)
2 123.456.78.9 - [25/Apr/2009:03:05:34 -0500] "GET carrier.html HTTP/1.0" 200 2050 A.html Mozilla/3.04 (Win95, I)
3 123.456.78.9 - [25/Apr/2009:03:05:39 -0500] "GET admission.html HTTP/1.0" 200 4130 - Mozilla/3.04 (Win95, I)
4 123.456.78.9 - [25/Apr/2009:03:06:02 -0500] "GET faciltiy.html HTTP/1.0" 200 5096 B.html Mozilla/3.04 (Win95, I)
5123.456.78.9 - [25/Apr/2009:03:06:58 -0500] "GET home.html HTTP/1.0" 200 3290 - Mozilla/3.01 (X11, I, IRIX6.2, IP22)
6123.456.78.9 - [25/Apr/2009:03:07:42 -0500] "GET carrier.html HTTP/1.0" 200 2050 home.html Mozilla/3.01 (X11, I, IRIX6.2, IP22)
7123.456.78.10 - [25/Apr/2009:03:07:55 -0500] "GET cse.html.html HTTP/1.0" 200 8140 admission.html Mozilla/3.04 (Win95, I)
8 123.456.78.10 - [25/Apr/2009:03:09:50 -0500] "GET placement.html HTTP/1.0" 200 1820 home.html Mozilla/3.01 (X11, I, IRIX6.2, IP22)
9 123.456.78.11 - [25/Apr/2009:03:10:02 -0500] "GET hostel.html HTTP/1.0" 200 2270 facility.html Mozilla/3.04 (Win95, I)
10123.456.78.11 - [25/Apr/2009:03:10:45 -0500] "GET course.html HTTP/1.0" 200 9430 placement.html Mozilla/3.01 (X11, I, IRIX6.2, IP22)
11 123.456.78.11 - [25/Apr/2009:03:12:23 -0500] "GET counselling.html HTTP/1.0" 200 7220 Carrier.html Mozilla/3.04 (Win95, I)
12 123.456.78.11- [25/Apr/2009:05:05:22 -0500] "GET home.html HTTP/1.0" 200 3290 - Mozilla/3.04 (Win95, I)
13 123.456.78.11 - [25/Apr/2009:05:06:03 -0500] "GET examination.html HTTP/1.0" 200 1680 home.html Mozilla/3.04 (Win95, I)

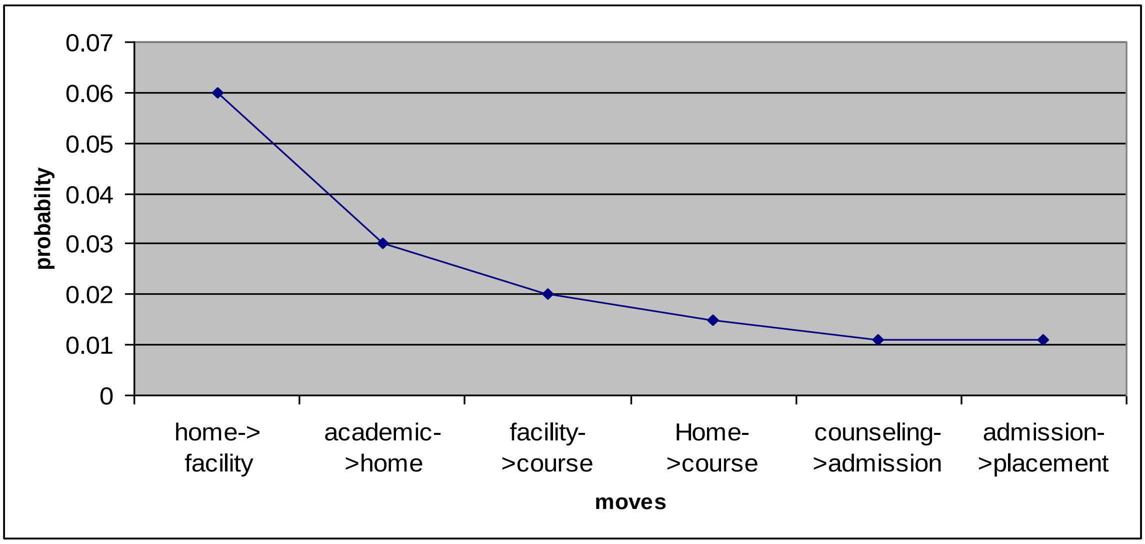

Figure 1.7: Zipf Curve

VI. CONCLUSIONS AND FUTURE SCOPE

Prefetching is characterized as one of the most efficient schemes to further reduce the user access latency, but runs the risk of increasing network traffic. Most of the approaches attempt to prefetch web objects according to some kind of criteria. The Web page access prediction accuracy can be improved by integrating all three prediction models: Markov model, Clustering and association rules according to certain constraints. After that Zipf Estimator can be applied on the rule generated from the previous phase. Our result allow a web prefetcher to take the advantage of Zipf law to accurately determine the probability of next page to be accessed . Efficient prefetching is very important for prefetching the next page. Zipf law holds the promise of more effective use of network resources.

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 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.

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

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

[4]. COOLEY, R., TAN, P-N., AND SRIVASTAVA, J. 1999b, “WebSIFT: The web site information filter system. In Proceedings of the Web Usage Analysis and User Profiling” Workshop (WEBKDD’99), Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Boston, August).

[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]. J. Wang, “A survey of web caching schemes for the internet,” ACM Computer Communication Review, vol. 29, no. 5, pp. 36–46, 1999.

[7]. M. Eirinaki and M. Vazirgiannis, “Web mining for web personalization,” ACM Trans. Inter. Tech., Vol. 3, No. 1, pp. 1-27, 2003.

[8]. COOLEY, R., MOBASHER, B., AND SRIVASTAVA, J. 1999a. “ Data preparation for mining world wide web browsing patterns”, Knowl. Inf. Syst., 1, 1 (Feb).

[9]. Lei Shi,Zhimin Gu,Lin Wei,and Yun Shi, “An applicative study of zipf’s law on web cache” ,In International Jounal of information Technology,Vol. 12 No.4 2006.

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

[11]. 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.

[12]. 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.

[13]. Marc Abrams et al. "Caching Proxies: Limitations and Potentials," http://ei.cs.vt.edu/~succeed/WWW4/WWW4.html

[14]. Jiang Y, Wu M, Shu W. "Web prefetching: costs, benefits and performance", In: Proceedings of the 7th international workshop on web content caching and distribution (WCW2002). Boulder, Colorado; 2002.

[15]. Sankar K. Pal,, Varun Talwar, and Pabitra Mitra , "Web Mining in Soft Computing Framework:Relevance" , State of the Art and Future Directions ,IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 13, NO. 5, SEPTEMBER 2002, page:1163-1177.

[16]. J. Hipp, U. Guntzer, and J. Nakhaeizadeh, "Algorithms for association rule mining a general survey and comparison," ACM SIGKDD Explorations, vol. 2, pp. 58–65, July 2000.

[17]. B. Mobasher, N. Jain, E.-H. Han, and J. Srivastava, "Web Mining: Patterns from WWW Transactions," Dept. Comput. Sci., Univ. Minnesota, Tech. Rep. TR96-050, Mar. 1997.

[18]. Jyoti Pandey ,Amit Goel, Dr. A K Sharma, A Framework for Predictive Web Prefetching at the Proxy Level using Data Mining, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008.