International Journal of Information Technology and Knowledge Management
July-December 2011, Volume 4, No. 2, pp. 343-345

PREPROCESSING ALGORITHM OF PREDICTION MODEL FOR WEB CACHING AND PERFECTING

Dharmendra Patel[1], Atul Patel[2] & Kalpesh Parikh[3]

Internet is boon in modern era, but sometimes people are unhappy while accessing web pages in spite of proper bandwidth of internet connection. To solve above problem, present work introduced one prediction model which predicts sequences of web pages in advance and store all web pages in cache memory of proxy server when user starts a session and as a result access latency to access web pages can be reduced. This prediction model consists of several components to do correct prediction. The components of prediction models are Preprocessing, User Session Identification, Pattern Generation and Prefetching.This paper introduces preprocessing component of prediction model. In this paper algorithm of preprocessing work is described with result and comparison of proposed work is made among Markov model, Popularity based model and LRS model. The paper presents conclusion that other model clean some useful web pages with unnecessary pages while this proposed algorithm make sure of that thing.

Key words: Web Caching, Web Prefecthing, Preprocessing, Sessionization.

1. Introduction

When any person access internet, server stores many information of that transaction in server log file. There are many formats of server log files [3]. This proposed work assumes that server uses W3C Extended log file format to record log files. The data available in log file in row ASCII format. This proposed research process row data of log file by omitting unnecessary fields and web pages for the purpose of web caching and prefetching [7] so access latency to retrieve web page can be reduced. The present work includes new algorithm of preprocessing [4] of log files to get relevant information to web caching and prefetching application. This proposed algorithm is compared against Markov model, Popularity based model and LRS model [1][5]. In present work row log file is downloaded form [8] , which consists of transactions of 15 days. In proposed research Microsoft Visual log parser tool is used to parse raw log data according to W3C Extended log file format and to omit unnecessary fields according to proposed algorithm using query based language. Proposed algorithm uses threshold value based on previous research [2][6] to store number of web pages in advance in server but that is depends on the memory of server. In second section preprocessing component of prediction model is discussed with Algorithm and Infrastructure. In third section Testing and Result analysis is done. Fourth section includes Conclusion of this paper.

[1]Charusat University, Changa, INDIA
[2]Charusat University, Changa, INDIA
[3]Director, IntellisenseIt Pvt.Ltd., Ahmedabad, INDIA
E-mail: [1]dharmendrapatel.mca@ecchanga.ac.in, [2]atulpatel.mca@ecchanga.ac.in, [3]kalpeshpar@gmail.com

2. Preprocessing Component

Preprocessing is the most fundamental task of proposed prediction model. Results of preprocessing task can be used as an input for other tasks. One sample row log file is downloaded from internet to test algorithm of preprocessing task. The sample of this raw log file is available in figure-1.

Fig.1: Sample Raw Log File

2.1. Steps of An Algorithm

[1] Parse the raw log file according to delimiter (space) and convert it into appropriate fields of W3C Extended Log File format.

[2] For graphics and video files, if time passed by user is less than time threshold value, remove that entry from log file. (According to proposed research 6 minutes for graphics and 10 minutes for video file is used and that is based on previous study).

[3] Remove all other entries which have other than .html,.asp,.aspx,.php extensions.

[4] Remove all entries having codes other than 200,304 and 306 from log files.

[5] Remove log entry which does not have any URL in URL entry.

[6] Calculate the access count for each and every web page that can be cleaned till step-5.

[7] Calculate the threshold value using access count request that has highest value in that file. Threshold value can be selected depends on memory of proxy server. Here in this proposed research, threshold value = (highest value*0.10) is selected which is based on previous research.

[8] If access count of any page > threshold value (Which is derived from above step) store that entry otherwise remove it.

2.2 Infrastructure Used For Preprocessing Component

- Personal Computer: Intel Pentium 4 CPU,2.40 GHz, 1 GB of RAM,20 GB Hard Disk
- Microsoft Log Parser Tool: This tool is used in proposed research to implements steps of algorithm (like parsing, retrieving selected records based on threshold value etc).
- Operating System: Microsoft Windows XP professional version 2002, Service Pack-2.
- Unicode Image Maker: This tool is used to convert text data into image file for documentation purpose.
- Microsoft Excel: This tool is used for simple calculation purpose.

3. TESTING AND RESULT ANALYSIS

Microsoft Visual Log Parser tool is used to test the algorithm. Four tests are generated to get the result of above mentioned algorithm.

Test 1: Parsing of raw file into W3C Extended Form

Test 2: Remove unnecessary web objects (Apply algorithm of steps 2 to 5)

Test 3: To determine unique web objects and associated hit count (Apply Algorithm of Step 6).

Test 4: To remove web objects which does not fulfill the condition of threshold value (Apply step-7 and step-8).

Following table (i.g Table-1) describe overall result of proposed algorithm after accomplishment of above mentioned tests.

Table 1
Result Analysis of Proposed Preprocessing Process

Stage of Preprocessing	No. of Web Objects Retrieved
Initial Stage (Before Preprocessing)	5000
After Test 2	2990
After Test 3	490
After Test 4	120
Cleaned Object (%)	97.6

Models like Markov, Popularity based and LRS do not take video and audio files in consideration while preprocessing, and as a result sometimes necessary files may be cleaned by them. We also perform testing of their algorithms in our infrastructure and the result is as under (i.e Table-2)

Table 2
Result Analysis of Other Models Preprocessing Process

Stage of Preprocessing	No. of Web Objects Retrieved by Other Models
Initial Stage (Before Preprocessing)	5000
After Test 2	1114
After Test 3	132
After Test 4	27
Cleaned Object (%)	99.4

4. CONCLUSION

Result analysis shows that other models cleans 99.4 % web objects while proposed model cleans 97.6 % web objects and binary objects like audio and video are in consideration of proposed algorithm if they meet threshold value decided by an algorithm . In modern era people access video and audio files more compared to only text files. We can also adjust threshold value of proposed algorithm based on cache memory of proxy server.

REFERENCES

[1] Faten Khalil, Jiuyong Li and Hua Wang, "Integrating Markov Model with Clustering for Predicting Web Page Accesses", Toowoomba, Australia, March 2007.

[2] James Pitkow, Peter Pirolli, "Mining Longest Repeating Subsequences to Predict World Wide Web Surfing", Proceedings of USITS' 99: The 2nd USENIX Symposium on Internet Technologies & Systems.

[3] K.R. Suneetha, Dr.R.Krishnamoorthi, "Identifying User Behavior by Analyzing Web Server Access Log File",

IJCSNS International Journal of Computer Science and Network Security, 9, No.4, April 2009.

[4] Mohd Helmy Abd Wahab, Mohd Norzali Haji Mohd, Hafizul Fahri Hanafi, Mohamad Farhan Mohamad Mohsin, Data Pre-processing on Web Server Logs for Generalized.

[5] Shantha Jayalal, Chris Hawksley, Pearl Brereton, "Website Link Prediction Using a Markov Chain Model Based on Multiple Time Periods", International Journal Web Engineering and Technology, Volume-3-2007.

[6] V. N. Padmanabhan and J. C. Mogul, "Using Predictive Prefetching to Improve World Wide Web Latency", Computer Communications Review, 26, pp. 22-36, July 1996.

[7] Wei-Guang Teng, Cheng-Yue Chang, Ming-Syan Chen, "Integrating Web Caching and Web Prefetching in Client-side Proxies, IEE, Parallel and Distributed System, Volume 16, May-2005.

[8] http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html.