Industrial Engineering
& Management Systems
Vol 11, No 1, Mar 2012, pp.82-86
ISSN 1598-7248 | EISSN 2234-6473 |
http://dx.doi.org/10.7232/iems.2012.11.1.082

Pre-Processing of Query Logs in Web Usage Mining

Norhaiza Ya Abdullah*
Malaysia Institute of Information Technology (MIIT), Universiti Kuala Lumpur
Husna Sarirah Husin
Malaysia Institute of Information Technology (MIIT), Universiti Kuala Lumpur
Herny Ramadhani
Malaysia Institute of Information Technology (MIIT), Universiti Kuala Lumpur
Shanmuga Vivekanada Nadarajan
Malaysia Institute of Information Technology (MIIT), Universiti Kuala Lumpur

(Received: November 25, 2011 / Revised: February 14, 2012 / Accepted: February 15, 2012)

ABSTRACT

In For the past few years, query log data has been collected to find user's behavior in using the site. Many researches have studied on the usage of query logs to extract user's preference, recommend personalization, improve caching and pre-fetching of Web objects, build better adaptive user interfaces, and also to improve Web search for a search engine application. A query log contain data such as the client's IP address, time and date of request, the resources or page requested, status of request HTTP method used and the type of browser and operating system. A query log can offer valuable insight into web site usage. A proper compilation and interpretation of query log can provide a baseline of statistics that indicate the usage levels of website and can be used as tool to assist decision making in management activities.

In this paper we want to discuss on the tasks performed of query logs in pre-processing of web usage mining. We will use query logs from an online newspaper company. The query logs will undergo pre-processing stage, in which the clickstream data is cleaned and partitioned into a set of user interactions which will represent the activities of each user during their visits to the site. The query logs will undergo essential task in pre-processing which are data cleaning and user identification.

Keywords: Pre-Processing, Web Log, Web Usage Mining

* Corresponding Author, E-mail: norhaizaya@miit.unikl.edu.my

1. INTRODUCTION

Recently with the explosive growth of the amount of content on the Internet, it has become increasingly difficult for users to search and utilize information. Content providers it is also difficult to classify and understand their user's need. The traditional web search engines often return hundreds or thousands of results for a search, which is time consuming for users to browse. Thus, the process of handling multiple data by multiple users can be time consuming and not efficient.

Basically, there are three types of information that need to be handled in a web site: content, structure and log data (Batista *et al*., 2002; Dixit and Gadge, 2010; Nicholas *et al*., 2004). In this paper, we concentrate on web usage mining, which is also known as web log mining. Web usage mining of query logs could help organization in understanding the patterns and profiles of their customer.

Web usage mining can be defined as automatic discovery and analysis of pattern of user access from web server (Cooley *et al*., 1999). Processes of pattern analysis in web usage mining are divided into three phases. The phases include preprocessing, pattern discovery, and pattern analysis. In preprocessing stage, the query logs are cleaned, users are identified and identification of session. Next, techniques of web usage mining such as association and clustering are performed to obtain

hidden patterns to discover the user behavior and profiles. For the last stages of pattern analysis, the discovered patterns are further processed resulting in aggregate user models used as input to generate a recommender tool.

In this paper, we present on the task of prepro-cessing in query log of web usage mining. For this project, we will use query logs from an online news-pa-per company. The query logs will undergo pre-proce-ssing stage, in which the clickstream data is cleaned and partitioned into a set of user interactions which will represent the activities of each user during their visits to the site. The rest of the paper is organized as follows: In section 2, we review some literatures in web usage min-ing and web log. Section 3 describes the implementation of preprocessing process which includes the preproces-sing algorithm. Results are shown in section 4 and section 5 acknowledge persons that permitting us to use the web server logs for the purpose of this study and finally section 6 summarizes the paper and future work.

2. RELATED WORK

Recently, many researchers are interested in web usage mining area. Web mining is the process of extracting knowledge from artifacts and activity related to World Wide Web (Cooley *et al*., 1999). Based on several studies, Web Usage Mining can be used for different purposes such as personalization, system improvement and site modification (Kumari and Raju, 2010). Data preprocessing phase is a challenging and difficult stage (Cooley *et al*., 1999). Data pre-processing stage is the most important phase for investigation of the web user usage behavior. To do this one must extract the only human user accesses from web log data which is critical and complex.

According to Cooley *et al*. (1999) the servers monitor such logging information and maintain the details using special log files. These files however represent information in form of raw textual data which is very difficult for the users to understand.

A Web server log is an important source for performing Web Usage Mining because it explicitly records the browsing behavior of site visitors. The data recorded in server logs reflects the access of a Web site by multiple users (Markellou *et al*., 2005). The greatest advantage of the Web server logs is that they are records of what people have actually done, and not what they might do or thought they did (Tyagi *et al*., 2010). The primary function of these logs are to record the operation of the web server, as well as for characterization, evaluation, reporting and website improvement (Mobasher *et al*., 2000; Nicholas *et al*., 2004). The web server logs are also commonly used to conduct analysis for the purposes of reporting traffic patterns for advertising or customer analysis. All log files are generated using the common log file format that several WWW servers use (w3c, 1995). Basically, log file contains information on each page request made to the web server. Figure 1 illustrates the format used in Berita Harian's logs. The information on the log starts with date and time, time zone of web server, IP address of clients, HTTP request status, cache size, URL or page requested, HTTP status code and user agent.

```
***15/Dec/2010:10:57:51 800 218.208.102.34 HIT Cache-Control: max-age=30000
        Expires: - GET /bharian/articles/NikAzizsifatketuananMelayusia-
   sia/Article HTTP/1.1   -200     Mozilla/4.0 (compatible; MSIE 8.0;
   Windows NT 5.1; Trident/4.0; TCO_20101215095625; .NET CLR 2.0.50727;
   .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; AskTB5.6)
***15/Dec/2010:10:57:51 800 202.86.248.231 EXPIRED Cache-Control:    -
        Expires: - GET /bharian/Ads/Images/iw234x74.gif HTTP/1.0     -200
        Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; FunWebProducts;
   SV1; GTB6.6; .NET CLR 1.1.4322; .NET CLR 2.0.50727; InfoPath.1; .NET CLR
   3.0.4506.2152; .NET CLR 3.5.30729)
***15/Dec/2010:10:57:51 800 60.52.224.133 HIT       Cache-Control:    -
        Expires: - GET /bharian/Gallery/imgFoto/sosilawati HTTP/1.1 -304
        Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0;
   InfoPath.2)
***15/Dec/2010:10:57:52 800 202.186.153.8  MISS     Cache-Control: no-store
        Expires: - GET /a_val HTTP/1.0     -200     KeepAliveClient
***15/Dec/2010:10:57:53   800 180.73.103.170 HIT    Cache-Control:    -
        Expires: -       GET /bharian/Images/bgvars.gif HTTP/1.1    -404
        Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0;
   GTB6.6; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR
   3.0.30729; InfoPath.2)
```

Figure 1. Extract of Server Logs from Berita Harian.

3. DATA PREPARATION AND PREPROC-ESSING

3.1 Web Server

Basically, several pre processing tasks need to be done before implementing web mining algorithm on web server logs. There are five preprocessing tasks as illustrated in Figure 2. The tasks are data cleaning, user identification, session identification, path completion and transaction identification (Cooley *et al*., 1999). To prepare the web server log for mining process, the data needs to be cleaned and preprocessed. Data cleaning is an important stage in data preprocessing. In data cleaning, certain techniques are used to remove irrelevant and non-significant items from the web server logs. In this project, the following are the steps of data cleaning.

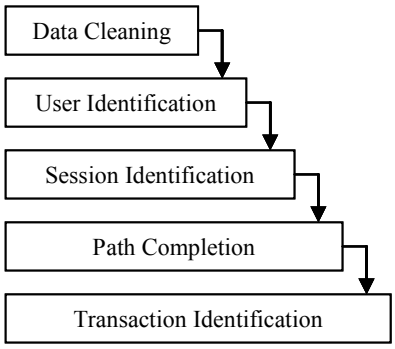

Figure 2. Preprocessing Process (Cooley *et al*., 1999)

- Step 1: Format the data. The data is retrieved from Nginx web server. It does not follow the conventional Common Log File (CLF) and Extended Log File (ELF) format.
- Step 2: Remove image files such as .jpg, .gif, .css and

all folders contain images
- Step 3: Remove HTTP status code other than 200. Status code 200 denotes as the request is successful. Other HTTP status codes found are 302, 304 (Not modified) and 404 (Not found).
- Step 4: Remove request method other than GET and POST. HEAD request method is considered irrelevant because it returns only headers in answer without content (Nicholas *et al*., 2004). Other request method such as PUT, DELETE, TRACE, CONNECT may contain bad request, properties of the server or visits of robots.

After the data has gone through extensive data cleaning, the next step is to identify user. A user can be defined as someone trying to access the web pages from the web server. In this paper, the following rules are observed (Dixit and Gadge, 2010).
- New IP address indicates new user
- If there is same IP Address, but the log files show different user agent, it represents new user.

3.2 Cache Server

Berita Harian uses cache server to expedite service requests by clients. This can be achieved because the cache server keeps local copies of frequently requested resources. If a user re-request the same data from the server, the cache re-send the same answer without requesting the server. The goal of caching to eliminate the need to send requests in many cases, and to eliminate the need to send full responses in many other cases. In web server logs, the cache status is either indicated by HIT, MISS, EXPIRED, UPDATING or STALE. HIT means that the page requested is available in the cache; MISS means that the request in not available in cache and to be read from the web server, EXPIRED happens when the cache age has expired. The use of cache server may cause problems of underreporting of site traffic, loss of referring site information and identifying site's usage. Proxy level caching could also cause a single request to be viewed by multiple users throughout an extended period of time. Consequently, user session identification will be difficult, because it is an arduous task to determine when the user's session is actually over (Srivastava *et al*., 2000).

3.3 Preprocessing Algorithm

The following is the extract of algorithm for preprocessing, done in Python 2.6. The web server logs are given to us are in the .tar format. The first step taken was to compile the web server logs based on the format. Most log files have their own unique characteristics format. As for this web server logs, we standardized the format according to date, time, time zone, IP address, cache status and cache control. Once the format is ready, we search the HTTP request based on Nginx HTTP log, which are method, path, protocol, status and browser. Then, the first stage of preprocessing of data cleaning; remove unnecessary image files. Here, we used regular expressions to remove all image files in the page request. Sometimes, the images are saved in folders, because Berita Harian always have gallery of images for their special content such as election pages, special events like Election, World Cup, images for button ads and many more. Once the images have been removed, the .tar file is parsed and put into a new database. The last step is to display the results in graph. The results are divided to status codes, cache status, HTTP method, browser and operating system of users.

Step 1: Compile the log file based on format desired; which is date, time, time zone, IP address, ca-che status and cache control

```
Log_Line1 = re.compile (
R'(?P <ts> (?P <date> \d{2}\/\w{3}\/\d{4}): (?P <time>
\d+: \d+: \d+)) (?P <tz> [\-\+]?\d\d\d)'
+ r '(?P <ip> \d+\.\d+\.\d+\.\d+) (?P <cache_status> \-
|MISS|EXPIRED|UPDATING|STALE|HIT) (?P
<cache_control> Cache-Control: [-|=|, |\w*]+)'
)
```

Step 2: Search HTTP request based on Nginx HTTP log; which is method, path, protocol, status and browser.

```
Log_Line2 = re.compile (
R ' "(?P <method> \w+) (?P <path> [\S]+) (?P <protocol>
[^"]+)" (?P <status> \(\d+\)) (?P <browser> \ "[^\"]*\ ")'' '
)
```

Step 3: Define the regular expressions to remove all images in the page request

```
pix_regex = re.compile
(r '(.png|.PNG|.gif|.GIF|.jpg|.JPG|.jpeg|.JPEG|.js|
.JS|.ico|.ICO|\/thumbnail|
\/pix|\/Pix|\/Images|.css|\/img|\/Ads|\/Gallery)')
```

Step 4: Read the log files. The log files is in the format of .tar

```
def readlog (filename, conn):
    m1 = m2 = z = w = c = 0 # just for error checkin
    conn = conn
    f = gzip.open(filename)
```

Step 5: Put each entry of log file into database named as totallog

```
query = "insert into totallog(total) values (?)"
    conn.execute (query, (w,))
    Break
```

Step 6: Count each HTTP status code (200, 302, 304, 404), count each cache status (MISS, HIT, EXPIRED), count each HTTP method (GET, POST, HEAD), count each browser (Internet Explorer, Firefox, Mozilla, Safari, Chrome, others), count each operating systems (Windows, Linux) display in chart

```
count = {'status_code': ['(200)', '(302)', '(404)', '(304)'], \
'cache_status': ['MISS', 'HIT', 'EXPIRED'], \
'http_method': ['GET', 'POST', 'HEAD'], \
'browser': ['Internet Explorer,' 'Firefox,' 'Chrome,'
'Opera,' 'Safari'], \
'OS': ['Windows,' 'Linux,' 'Macintosh,' 'Iphone']}
```

4. RESULTS

An experiment using web server logs was conducted to test our algorithm. For this experiment, we used 750MB of data, which results to 401809 entries of logs. The first step in our data cleaning stage is to remove all images, which include .gif, .jpeg, and .css. Due to the format of the log file, some of the images are hidden in folder. Therefore, the log file has to be examined carefully to find image folders, as well as the image files. Table 1 shows that the number of log files has considerably decrease after all the images are removed, from 401809 to only 44014, which constitutes approximately 11% of the original data.

Table 1. Results of Log Files After Removing Image Files.

Number of web logs before preprocessing	Number of web logs after data cleaning
401809	44014

After the images are removed, the next step is to filter the status code. Figure 3 shows the different status codes were identified, and as a result, only status code of 200 is used.

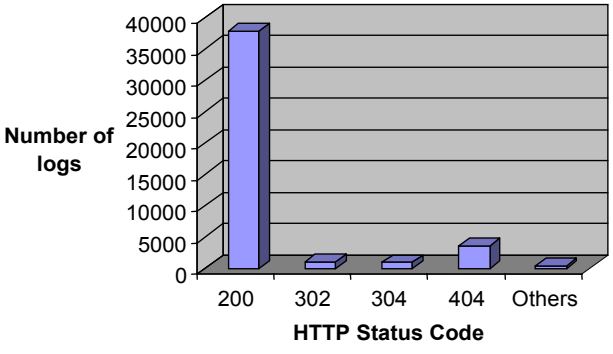

Figure 3. Web Server Logs Based on HTTP Status Code.

Figure 4 is the results of user identification. After the IP address of each user is identified, the users are further divided into different user agents. This is based on the rules states that if the IP address is the same, but user agent is different, then it denotes different user. From the graph, the highest user agent is Internet Explorer, followed by Firefox, Chrome, Opera and Safari. Other user agents include accesses from browsers used in wireless devices such as smart phone, iPhone or Blackberry. Figure 5 illustrates the user identification based on list of IP address, the different browser of each IP, and the page requested. In this figure, user is identified based on their IP address. Although there are many same IP address, but if the page is accessed from different browser, it shows that they are different users.

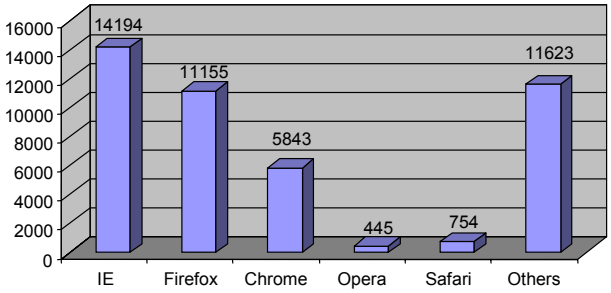

Figure 4. User Identification Based on Browser.

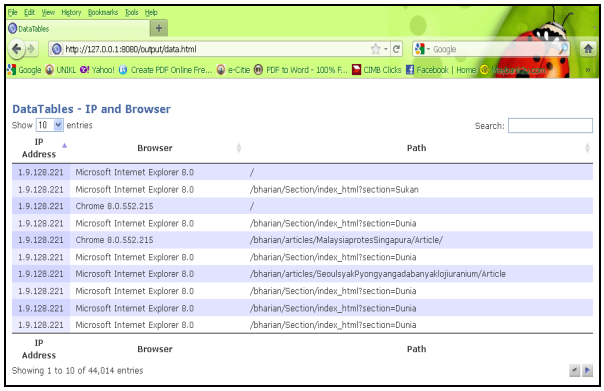

Figure 5. Results of IP Address, Browser and Page Request.

5. CONCLUSION

In this paper, we presented our detailed of preprocessing phase, which is used to clean web server logs. By using our script in Python 2.6, we define the regular expressions and provide rules for every requirement we need to clean. The experiment conducted has successfully cleaned the web server logs from unnecessary and non-significant information. The testing from the script shows the importance of preprocessing phase as it not just reduce the log file size, as well as increase the quality of available data, which will be used in the pattern discovery phase in the web usage mining phase later. Moreover, there are still issues that need to be resolved such as identifying session and transactioni-

zation. Future study will identify appropriate measure to session the data, due to the fact that cache server is used to access the most recent page request by the client.

ACKNOWLEDGEMENT

The research team thanks Bakhtiar Abdul Hamid from New Straits Times Press for assisting to retrieve the web server logs and Zainudin Mohd Isa, Editor of Berita Harian Online for permitting us to use the web server logs for the purpose of this study.

REFERENCES

Batista, P., Silva, M. J., Silva, M., and Grande, C. (2002), Mining On-line Newspaper Web Access Logs, Proceedings of the AH'2002 Workshop on Recommendation and Personalization in eCommerce, 100-108.

Choa, Y. H., Kim, J. K., and Kima, S. H. (2002), A personalized recommender system based on web usage mining and decision tree induction, *Expert Systems with Applications*, **23**, 329-342.

Cooley, R., Mobasher, B., and Srivastava, J. (1999), Data Preparation for Mining World Wide Web Browsing Patterns, *Knowledge and Information Systems*, **1**(1), 5-32.

Dixit, D. and Gadge, J. (2010), Automatic Recommendation for Online Users Using Web Usage Mining, *International Journal of Managing Information Technology (IJMIT)*, **2**, 33-42.

Elsheikh, S. (2008), Web Usage Data for Web Access Control (WUDWAC), Proceedings of the World Congress on Engineering.

Hao, T., Brimmer, D. J., Lin, J. M. S., Tumpey, A. J. and Reeves, W. C. (2009), Web Usage Data as a Means of Evaluating Public Health Messaging and Outreach, *Journal of Medical Internet Research*, **11**, 99-118.

Vellingiri, J. S. And Pandian, C. (2011), A Survey on Web Usage Mining, *Global Journal Of Computer Science and Technology*, **1**, 4343-4350.

Kumari, V. V. and Raju, K. S. (2010), Understanding User Behavior using Web Usage Mining, *International Journal of Computer Applications*, **7**, 162-286.

Markellou, P., Rigou, M., and Sirmakessis, S. (2005), Mining for Web Personalization, in Scime, A. (Ed.) *Web Mining: Applications and Techniques*, London: Idea Group Publishing, 27-48.

Mobasher, B., Dai, H., Luo, T., Sun, Y., and Zhu, J. (2000), Integrating web usage and content mining for more effective personalization, Proceedings of the First International Conference on Electronic Commerce and Web Technologies, LNCS, **1875**, 165-176.

Murgue, T. and Jaillon, P. (2005), Data Preparation and Structural Models for Web Usage Mining, *SETIT International Conference: Sciences of Electronic, Technologies of Information and Telecommunication*.

Nicholas, D., Huntington, P., Williams, P., and Dobrowolski, T. (2004), Reappraising information seeking behavior in a digital environment, *Documentation*, **60**(1), 24-43.

Pitkow, J. (1997), In search of reliable usage data on the WWW, Sixth International World Wide Web Conference, 451-463.

Srivastava, J., Cooley, R., Deshpande, M., and Tan, P. N. (2000), Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data, *ACM SIGKDD*, **1**(2), 12-23.

Sanjay, B. and Thakare, S. (2010), A effective and complete preprocessing for Web Usage Mining, *IJCSE International Journal on Computer Science and Engineering*, **2**(3), 848-851.

Status codes (2011), Available at http://www.w3.org/Protocols/HTTP/HTRESP.html.

Tanasa, D. and Trousse, B. (2004), Advanced Data Preprocessing for Intersites Web Usage Mining. *IEEE Intelligent Systems,* **19**(2), 59-65.

Tyagi, N. K., Solanki, A. K., and Wadhwa, M. (2010), Analysis of Server Log by Web Usage Mining for Website Improvement, *International Journal of Computer Science Issues*, **7**(4-8), 17-21.