Knowledge Discovery from Web Usage Data: An Efficient Implementation of Web Log Preprocessing Techniques

Shivaprasad G.
Manipal Institute of Technology,
Manipal University,
Manipal

N.V. Subba Reddy
Manipal Institute of Technology,
Manipal University,
Manipal

U. Dinesh Acharya
Manipal Institute of Technology,
Manipal University,
Manipal

ABSTRACT
Web Usage Mining (WUM) refers to extraction of knowledge from the web log data by application of data mining techniques. WUM generally consists of Web Log Preprocessing, Web Log Knowledge Discovery and Web Log Pattern Analysis. Web Log Preprocessing is a major and complex task of WUM. Elimination of noise and irrelevant data, thereby reducing the burden on the system leads to efficient discovery of patterns by further stages of WUM. In this paper, Web Log Preprocessing Methods to efficiently identify users and user sessions have been implemented and results have been analyzed.

General Terms
Data Mining, Web Usage Mining

Keywords
Web Usage Data, Web Log Pre-processing.

1. INTRODUCTION
Web mining is the application of data mining techniques to extract knowledge from the huge amount of information available on the web. It refers to the effort of Knowledge Discovery in Data (KDD) from the web.

Based on the data to be mined, Web mining can be categorized into three major areas [1]:

- *Web Content Mining* deals with the discovery of useful information from the web contents or data or documents or services.
- *Web Structure Mining* deals with discovery of useful information from the structure of hyperlinks within the web itself. Structure represents the graph of the link in a site or between the sites.
- *Web Usage Mining* deals with extraction of useful information from the log data stored in the web server.

The proposed research, mainly concentrates on Web usage mining as a means to track the behavioral patterns of users surfing either a web site or a page. Web Usage mining consists of three main steps: Preprocessing, Knowledge Discovery and Pattern Analysis.

Mining for knowledge from Web log data has the prospective of revealing useful information. In general context, Web Usage Mining is employing Data Mining algorithms over the Web Usage Data. However, this process is not just adapting existing algorithms to new data. The WUM algorithms take as input the raw web log which is a rich repository of user activities on the web. A user session represents the sequences of page accesses that a user does in his visit to a Web site.

The information contained in a raw web server log does not reliably represent a user session file. Web log data is unstructured in format and consists of ambiguous and irrelevant data and preprocessing the log data is an important and essential step before application of mining. Generally, several steps are involved in preprocessing the raw server logs. These include eliminating irrelevant items, identifying unique users and user sessions within a server log. Preprocessing of Web log data is a complex task and consumes 80% of overall mining process [2]. The important steps of web log data preprocessing are identified as pruning noisy and irrelevant data and to reducing volume of log data for the further pattern discovery phase. The data mining algorithms can then be applied to preprocessed log data to extract patterns which are then analyzed to reveal useful information.

In this study, the Web Log Preprocessing Methods to efficiently identify users and user sessions have been implemented. The paper is organized as follows: a brief related work, web usage data and web log preprocessing, Web Log Preprocessing Algorithms, Experimental Setup and results and conclusions and future works.

2. RELATED WORK
Web Log Preprocessing plays a major role in WUM. The raw log data doesn't suitable for Data Mining algorithms. Also, the size of the data is huge and demands memory and processor. The Web Log Preprocessing should produce reliable and good quality data so that the Data Mining algorithms can produce useful patterns. The Web Log Preprocessing algorithms should results in meaningful user sessions which are then analyzed by the Data Mining algorithms.

According to [3], web logs can predict user's next request without disturbing them. However, the not all details/files available in web logs are appropriate for mining navigation patterns. So the information from web logs needs cleaning before it can be used for prediction. The main objective is to find only the valid and frequently requested HTML documents. Therefore, unnecessary files are deleted using cleaning algorithm. In [4], several data preparation techniques used to improve the performance of the data preprocessing to identify the unique sessions and unique users is presented. A Field extraction algorithm to separate the fields from the single line of the log file and a Data cleaning algorithm to eliminate inconsistent or unnecessary items in the analyzed data is presented in [5]. Preprocessing techniques based on some heuristics is proposed in [6]. [7] and [8] discusses in

detail the various Web log preprocessing algorithms employed in WUM. In [9], an intelligent algorithm for Web Log Preprocessing is proposed.

3. WEB USAGE DATA AND WEB LOG PRE-PROCESSING

The main objectives of Web Log Preprocessing are to reduce the quantity of data being analyzed while, at the same time, to enhance its quality. Preprocessing mainly comprises of the following steps – Collection of Data, Data Cleaning, User Identification and Session Identification.

3.1 Web Usage Data

Web servers automatically collect the data of user activities on web. This Web Usage Data represents the accurate navigational behavior of visitors. It is the primary source of data in WUM. Each hit against the server, corresponding to an HTTP request, generates a single entry in the server access logs. A Web server log generally contains the request made to the server. In Common Log Format (CLF) the important fields of web log are identified as Host, Username, Password, Date and time of Request, HTTP Request, Status of the Request, Page size, Referrer of the Request and User agent. Each of these fields is detailed below:

- Host/The Remote IP address: This identifies who had visited the web site.
- User Authentication: Username and password if the server requires user authentication (generally empty and represented by a "-").
- Date and time of the request: This attribute is used to determine how long a visitor has spent on a given page.
- The HTTP Request: The method (GET, POST, HEAD, etc.) used for information transfer is noted along with the requested resource name (an HTML page, a CGI program, or a script) and protocol Version (HTTP protocol being used).
- The Request status: The HTTP status code returned to the client (200, 404 etc.).
- The Page size: The content-length of the document transferred.
- The Referrer: This lists the URL of the previous site visited by the client, which is linked to the current page (- if this information is missing).
- The User agent: This provides the information about the client's browser, the browser version, and the client's operating system (- if this information is missing).

A sample HTTP request for the NASA Kennedy Space Centre WWW server [10] is depicted in figure 1.

```
199.72.81.55 - - [01/Jul/1995:00:00:01 -0400]
"GET /history/apollo/ HTTP/1.0" 200 6245
```

Fig. 1: An extract of log record from NASA web log.

The web access log line in Figure 1 shows that a user from the IP address 199.72.81.55 successfully requested the page "apollo" on 01st July 1995 at 00:00:01 a.m. The user has used the HTTP method, "GET" to access the page and a total of 6245 bytes were returned.

3.2 Web Log Pre-Processing Steps

Web Log Preprocessing is a complex and time consuming task. In this section, the various steps for transforming a raw web log data into a form that is useful further analysis by data mining algorithms are discussed.

3.2.1 Field Separation

Usually, in a raw web log, data pertaining to multiple fields is represented using a single field. The page access date and time is represented using the date/time field as a single entity. Similarly, the request type, requested resource and the protocol are together represented as one entry in HTTP request field. For further analysis, it may be useful if these combined data is represented separately. Hence, in this step, the Date/Time, HTTP request and User agent fields are analyzed and fields are separated.

3.2.2 Time- Stamp Computation

For analyzing the web log, the time duration a user spends in website need to estimated using the Date/Time field of web log. For this purpose, the Time stamp, which represents the number of seconds elapsed since the midnight of the base date is computed.

3.2.3 Data Cleaning

Web log consists of redundant requests and irrelevant entries in addition to user actual requests. Elimination of these unwanted data plays a vital role in Web Log Preprocessing. Data Cleaning speeds up the upcoming data mining tasks by reducing the size of the weblogs. A user request to a specific page from server, usually results in additional entries like gif, JPEG, etc. in weblog, as these files are also downloaded. These entries can be eliminated as they are not useful for the analysis by the Data Mining algorithms. Also, the web log must be freed from the requests from automatic software agents, crawlers and bots so as to represent actual user access pattern. In addition to the successful requests, the web log contains the records that were unsuccessful, which need to be removed. Thus, data cleaning includes the following sub steps:

3.2.3.1 Removal of robot Requests

Search engines are interested in giving the most current information to their visitors. They gather this information by sending software agents, robots or spiders to World Wide Web. These bots crawl around the web exploring all possible links. However, WUM is intended to analyze the user access behavior. Hence, all entries pertaining to non-human access has to be eliminated from the web log.

Usually standard robots access *robots.txt* file before downloading any other documents. In some cases, they reveal their identity in the User agent field to announce their presence. Hence, the User agent field is analyzed to remove the log records of web robots.

3.2.3.2 Removal of Image Requests

When web pages consisting of images are accessed, the log will have entries representing the request for image files in addition to the web page visited. The WUM is interested in analyzing user's actual access requests. Hence, the log records representing image accesses and other associated data have to

be eliminated. This cleaning process helps to remove redundant data thereby speeds up the rest of the process.

3.2.3.3 Removal of Unsuccessful HTTP Method

The HTTP status code of a web log record represents whether the request for the page was successful or unsuccessful. Records with only successful requests (status code = 200) are retained. All the other records with unsuccessful requests can be eliminated. This step reduces the size of the web log thereby avoids the burden on the memory and speeds up the analysis.

3.2.3.4 Removal of non-GET Methods

As WUM is mainly interested in analyzing the user access patterns, web log records having the value of GET in Method field are retained and other records are removed.

3.2.4 User Identification

Identification of each distinct user is the main aim of User Identification step. This step would be simple to achieve if each user is provided with separate login credentials. However, most user accesses to most of the Web sites are done incognito mode, so that registration information is optional/not available.

The user identification becomes a very complex task in the absence of user-id field in the access log(which, in most cases, is blank). Apart from the user-id field, the fields available for user identification are [11]:

- IP address
- User Agent
- Referring URL

User Identification Algorithm based on above 3 fields is proposed in [12].

3.2.5 User Session Identification

The sequence of activities performed by a user from the moment he enters the website to the moment he leave the website is referred to as a session. In User Session identification, the web access log of every user is split into sessions and these sessions are further analyzed.

As per [13], three heuristic methods based on time and navigation is usually considered for session identification. The *time-oriented heuristics* rely on session duration or page stay time. The *navigation-oriented heuristic* is based on user navigation pattern. Each heuristic scans the user activity logs and divides the user activity into sessions.

Session-duration based heuristic: The total duration of a session cannot exceed a threshold Θ. As per [14], a new request is considered as a new session if this request was given after the above threshold. As a benchmark, thirty minutes threshold is employed in many applications.

Page-stay-time based heuristic: Here, a threshold is employed as the maximum page stay time. As per [14], if the current request was done within the above threshold then this request is added to constructed session. A ten minutes threshold is employed in many applications.

Referrer-basic heuristic: In accordance with [14], a request q is added to constructed session S if the referrer for q was previously invoked in S; otherwise, q is used as the start of a new constructed session.

4. WEB LOG PRE-PROCESSING ALGORITHMS

The proposed Web Log Pre-processing mainly consists of following algorithms:

4.1 Generalized Web Log Cleaning algorithm

4.2 User Identification algorithm

4.3 Session Identification algorithm.

4.1 Generalized Web Log Cleaning Algorithm

The algorithm takes as input the web log file obtained after the field separation and Time-Stamp computation. IsRobot, IsUnwaanted, IsSuccRequest and IsNonGet are Boolean functions which analyze the given log record as detailed in previous section. The algorithm is as follows:

Input: Web Log File obtained after Feature Extraction Method

Output: Cleaned Log File

Algorithm **WebLogCleaning**

```
Begin
  While not eof (WebLogFile) Do
        Read next record of WebLogFile into Log-Record
        If NOT(IsRobot(Log-Record)) and
          NOT(IsUnwanted(Log-Record)) and
          IsSuccRequest(Log-Record) and
          NOT(IsNonGet(Log-Record))
                Write Log-Record to New-LogFile.
        EndIf
  EndWhile
End
```

4.2 User Identification Algorithm

User Identification Algorithm based on IP-address is given below:

Input: N records of cleaned web log file.

Output: User set U

Algorithm **UserIdentification**

Begin

User-Count ←0

Sort the Log Records with respect to IP-Address

While not eof (New-LogFile) Do

(i) Read current record of New-LogFile into Cur-Record

(ii) Read next record of New-LogFile into Next-Record

(iii) Compare IP-Address of Cur-Record with IP-Address of Next-Record

(iv) If both IP-Address are same

Identify both entries as belonging to same user.

Else

Assume as different users and Increment the

Table 1. Results of Field Separation and Time-Stamp Computation Step

USERID	DATE AND TIME	METHOD	REQUESTED PAGE	PROTOCOL	STATUS	No. OF BYTES	TIMESTAMP
199.72.81.55	01-Jul-1995 0000:01	GET	/history/apollo/	HTTP/1.0	200	6245	1.16E-05
unicomp6.unicomp.net	01-Jul-1995 00:00:06	GET	/shuttle/countdown/	HTTP/1.0	200	3985	6.94E-05
199.120.110.21	01-Jul-1995 00:00:09	GET	/shuttle/missions/sts-73/mission-sts-73.html	HTTP/1.0	200	4085	1.04E-04
205.212.115.106	01-Jul-1995 00:00:12	GET	/shuttle/countdown/countdown.html	HTTP/1.0	200	3985	1.39E-04
d104.aa.net	01-Jul-1995 00:00:13	GET	/shuttle/countdown/	HTTP/1.0	200	3985	1.50E-04

User-Count by 1.

EndIf

EndWhile

End

4.3 Session Identification Algorithm

The time-oriented heuristic based session duration algorithms were employed for session identification. Here, if a user has accessed the web page for more than 30 minutes, this session will be divided into more than one session. The algorithm is as given below:

Input: User sets with N records

Output: Constructed Sessions SessionSet

$K \leftarrow K+1$

$S_k \leftarrow URI_j$

SessionSet = SessionSet U S_k

Else

If $(t_j - t_{j-1}) <$ SessionTimeout

Add URI_j to S_k

End If

EndIf

End For

End For

End

Algorithm **SessionIdentification**

Begin

Let SessionTimeout $\leftarrow$30 minutes

SessionSet = { }

$K \leftarrow 0$

Let L_j, URI_j, t_j and U_j denote log entry, URI, time stamp and user respectively.

For each unique user U_j do

For each Lj do

If $(t_j - t_{j-1}) >$ SessionTimeout

5. EXPERIMENTAL SETUP AND RESULTS

The experimental data represents the HTTP requests to NASA Kennedy Space Centre WWW server in Florida from 00:00:00 July 1, 1995 to 23:59:59 July 31, 1995, available at Internet Traffic Archive repository. A sample log consisting of 1,00,000 requests was used for experimental setup from the available repository. The sample log records were then pre-cleaned to eliminate 7 invalid entries, leading to a log file consisting of 99,993 records for Web Log Preprocessing. The algorithms were implemented in MATLAB.

5.1 Field Separation and Time-Stamp Computation

The features or fields like UserID, Date and Time, Request Method, Requested Page(URI) and Protocol are extracted from the web log record. Then the time stamp is computed using built-in function of MATLAB. The results of this step are shown in Table 1.

5.2 Data Cleaning

The Data Cleaning Step identified 54661 records containing multimedia objects, 7 corrupt requests and 10347 failed requests with a total of 39210 clean log records ready for further processing. The input log size is considerably reduced by this step. This analysis stresses the importance of Data Cleaning Step. The raw web log data contained approx. 60% of unwanted data. The Table 2 and 3 shows the statistics about individual request category and aggregated results of Data Cleaning Step. The statistics reveal that approx. 50% log records contained the request for multimedia objects. Also, it can be observed from the results that around 89% of the requests for the pages were successful.

Table 2. Statistics about individual request category

Request Category	Number of records	Percentage
Gif	50686	50.69
Jpeg	374	0.3
Jpg	3601	3.6
Css	0	0
Js	0	0
GET	99905	99.91
200(SUCCESSFUL)	89646	89.65

Table 3. Statistics about aggregate results of data cleaning

Statistics	Number of records
Original size	100000
Satisfied requests	89646
Corrupt requests	7
Failed requests	10347
Multimedia objects	54661
Cleaned log size	39210
Percentage in reduction	39.21

Table. 4. A fraction of log files with user sessions

USERID	Session Id
199.72.81.55	0
199.72.81.55	1
unicomp6.unicomp.net	0
205.212.115.106	0
205.212.115.106	1
205.212.115.106	2
205.212.115.106	3
205.212.115.106	4
205.212.115.106	5
205.212.115.106	6
205.212.115.106	7
205.212.115.106	8
205.212.115.106	9
205.212.115.106	10

The distribution of relevant and irrelevant data in Web Log after the processing by Data Cleaning is depicted in Figure 2. The results show that a major portion of raw log data contained the request for images. Such requests are actually the indirect requests resulted when accessing the web pages and hence can be eliminated to reduce the load on the processor.

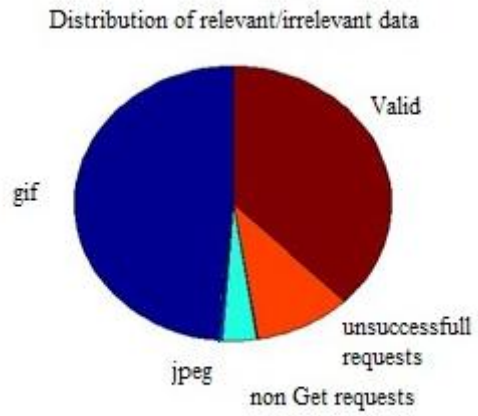

Fig.2. Distribution of relevant and irrelevant Data in Web Log

5.3 User Identification

The User Identification Algorithm uniquely identifies the users of the Web Site. A total of 7352 users were identified in the given log.

5.4 User Session Identification

The Session Identification splits all accesses by each user into individual access sessions using the time oriented heuristic. A fraction of the log file with user sessions is shown in Table 4.

6. CONCLUSION

Preprocessing of data is an essential activity which will help to improve the quality of the data and successively enhance the quality of mining results, as a result it enhances the performance of the system. Web Log Preprocessing is one of the important steps in Web Usage Mining. Data Cleaning step revealed that a major portion of Web Log usually consists of irrelevant and redundant data which has to be eliminated to speed up the upcoming mining process.

In this work, the Web Log files from NASA web server have been considered for preprocessing. An attempt is made to reduce size of the log file. The size of the log file reduced considerably to that of the original log size. Enhancing the quality of log data is also of utmost importance in addition to reducing the quantity of data.

Further, the Web Log Preprocessing algorithms need to be improved by considering the removal of remaining robots

requests and other multimedia files or irrelevant data. Further possible exploration is to address problems such as the accuracy metric of the user identification and the session identification, and applying the results of the preprocessing to discover useful patterns in the web mining process. Also, the structured session file needs to be transformed, either by using relational data model or data structures for further processing by data mining algorithms.

7. ACKNOWLEDGMENTS

The authors are thankful to Mr. Prakash K. Aithal, Manipal Institute of Technology, Manipal University for implementation of the algorithms in MATLAB.

8. REFERENCES

[1] Supriya Kumar De and P. Radha Krishna, 2004, "Clustering web transactions using rough approximation, Fuzzy Sets and Systems", Vol. 148, Science Direct, 131–138.

[2] Nitya P. and Sumathi P., 2012, "Novel pre-processing technique for web log mining by removing global noise and web robots", National Conference on Computing and Communication Systems (NCCCS), IEEE, 1-5.

[3] G. Arumugam and S. Suguna, 2009, "Optimal Algorithms for Generation of User Session Sequences Using Server Side Web User Logs", International Conference on Network and Service Security, IEEE, 1-6.

[4] Sudheer Reddy K., Kantha Reddy M. and Sitaramulu V., 2013, "An effective data preprocessing method for Web Usage Mining", International Conference on Information Communication and Embedded Systems (ICICES), IEEE, 7-10.

[5] Aye T.T., 2011, "Web log cleaning for mining of web usage patterns", Third International Conference on Computer Research and Development (ICCRD), Vol. 2, IEEE, 490-494.

[6] K Sudheer Reddy, G. Partha Saradhi Varma and I. Ramesh Babu, 2012, "Preprocessing the Web Server Logs – An illustrative approach for effective usage mining", ACM SIGSOFT Software Engineering Notes, Vol.37. No. 3, 1-5.

[7] G. Shiva Prasad, N.V. Subba Reddy, and U. Dinesh Acharya, 2010, "Knowledge Discovery from Web Usage Data: A Survey of Web Usage Pre-processing Techniques", Proc. of International Conference on Recent Trends in Business Administration and Information Processing (BAIP 2010), Communications in Computer and Information Science, Vol. 70, Springer Berlin Heidelberg, 505-507.

[8] Brijesh Bakariya, Krishna K. Mohbey and G. S. Thakur, 2013, "An Inclusive Survey on Data Preprocessing Methods Used in Web Usage Mining", Proc. of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012), Advances in Intelligent Systems and Computing Vol. 202, Springer India, 407-416.

[9] Zhang Huiying 2004, "An Intelligent Algorithm of Data Pre-processing in Web Usage Mining", Proceedings of the 5th World Congress an Intelligent Control and Automation, Vol. 4, IEEE, 3119 – 3123.

[10] NASA Kennedy space Centre http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html

[11] R. Suresh and R. Padmajavalli, 2007, "An Overview of Data Preprocessing in Data and Web Usage Mining", First International Conference on Digital Information Management, 193-198.

[12] G. Arumugam and S. Suguna, 2008, "Predictive Prefetching Framework Based on New Pre-processing Algorithms towards Latency Reduction", Asian Journal of Information Technology, Vol.7, No.3, 87-99.

[13] Das, R. and Turkoglu, I., 2009, "Creating meaningful data from web logs for improving the impressiveness of a website by using path analysis method", Expert Systems with Applications, Vol. 36, Science Direct, 6635–6644.

[14] Liu, B., 2007, Web data mining: Exploring hyperlinks, contents and usage data, Springer, ISBN: 13-978-3-540-37881-5 (532p.).