ISSN No. 0976-5697

Volume 5, No. 5, May-June 2014

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Improvisation in Web Mining Techniques by Scrubbing Log Files

Rachit Goel
Department of Computer Science and Engineering
M.tech Scholar, Doon Valley, Karnal, India

Sandeep Jain
Department of Computer Science and Engineering
Asst. Professor, Doon Valley, Karnal, India

Abstract **-** World Wide Web is one of the most interactive and popular medium to spread the information. The increasing popularity and size growth of WWW has overwhelmed with an immense amount of widely dispersed interconnected and dynamic information. Web pages typically contain a large amount of information that is not part of the main content of the pages, e.g. banner ads, navigation bars, copyright notices, etc. Such noise on web pages usually leads to poor results in Web Mining which mainly depends upon the web page content. Therefore, it becomes very essential to extract information from the bulks of data and structure them into useful knowledge that will be helpful for some type of understanding. This leads to the birth of data mining. Web usage mining is the subject field of Web mining which deals with the discovery and analysis of usage patterns from web data specifically web logs in order to improve the web based applications. The Web usage mining process consists of three phases: Data Preprocessing, Pattern Discovery and Pattern Analysis. In this paper an improvised algorithm will be proposed that gives a clean file consisting of only relevant data from the Web usage mining perspective as output.

Keywords- Data Mining, Web Mining, Web Usage Mining, Data Pre processing.

I. INTRODUCTION

The World Wide Web abbreviated as WWW and commonly known as the Web is a system of interlinked hypertext documents which can be accessed through the Internet. There are billions of web search engines (such as Google, Yahoo, Bing, etc.), that processes large amount of data. The list of sources that generate huge amounts of data is endless. However, the abundant information on the web is not stored in any systematically structured way, which poses great challenges to those looking for high quality information underlying in web pages. The growth in the mass of data present on web has engrossed the attention of the scholars and researchers towards the application of data mining techniques on the data available on the web in order to extract useful information. Web mining has therefore become an important subject matter in data mining.

Therefore, it becomes very essential to extract useful information from the bulks of data. According to Frawley, data mining is defined as "The nontrivial extraction of implicit, previously unknown, and potentially useful information from data" [1]. Data mining can also be defined as the process of discovering meaningful new correlation, patterns and trends by analyzing the large amounts of data, using pattern recognition technologies as well as statistical and mathematical techniques. Data mining is a step that consists of applying data analysis and discovery algorithms that produce a particular enumeration of patterns (or models) over the data [2]. Web usage mining is process of discovering usage patterns from Web data, in order to better understand the needs of web based applications. Web usage mining deals with the extraction of knowledge from web server log files. The main source of data for Web usage mining mainly consists of the (textual) logs, which are collected when users access web servers. A high level web usage mining process is shown in figure 1. The Web usage mining is parsed into three distinctive phases [3]:

a) ***Data Preprocessing-*** It performs a series of steps covering data cleaning, user identification, session identification, path completion and transaction identification.

b) ***Pattern Discovery-*** It involves application of various data mining techniques to processed data like statistical analysis, association, clustering and pattern matching.

c) ***Pattern Analysis-*** Filters out irrelevant patterns from the identified patterns generated in pattern discovery phase.

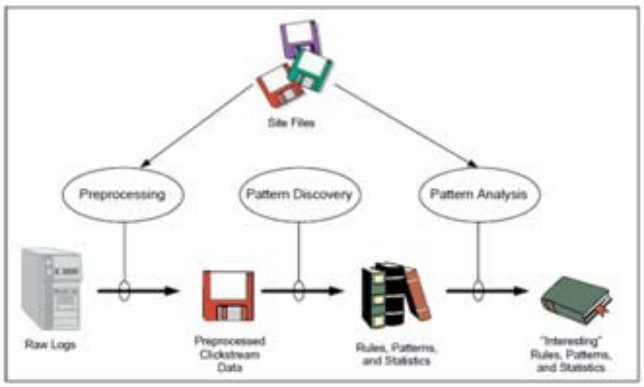

Figure. 1: A High Level Web Usage Mining Process [4]

II. WEB MINING TASKS

Web mining can be decomposed into the following subtasks [13] :

a) ***Information Retrieval (or Resource Discovery):*** Search is probably the one of the prime application of the Web. Information retrieval (IR) helps the users to find the required information available from a large collection of text documents.

b) ***Information Extraction (Selection and Preprocessing):*** This task deals with the transformation of the data retrieved during information retrieval process into a form that can be easily analyzed [7]. Information extraction aims to select relevant facts from the documents while information

retrieval aims to select relevant documents.

c) ***Generalization (Pattern Recognition and Machine Learning):*** It automatically generates general patterns from both the individual web sites as well as across multiple sites. Machine learning methods or data mining techniques are generally used for the generalization purpose.

d) ***Analysis (Validation and Interpretation):*** Once the patterns have been identified it is necessary to explore and confirm those mined pattern. The aim of this task is to validate the mined patterns.

Based on the above mentioned subtasks, web mining can be viewed as the use of data mining techniques to automatically retrieve, extract and evaluate information for knowledge discovery from web documents and services [6].

III. WEB MINING TYPES

The advent of the World Wide Web has made the data present on the web as a gigantic source of information. The World Wide Web, having over 350 million pages, continues to grow rapidly at a million pages per day [4]. This increase in the mass of data has turned researcher's attention towards the use of data mining techniques to extract useful information from the web data.

Web mining can be classified into three types [5] as shown in figure 2:

a) Web Content Mining
b) Web Structure Mining
c) Web Usage Mining.

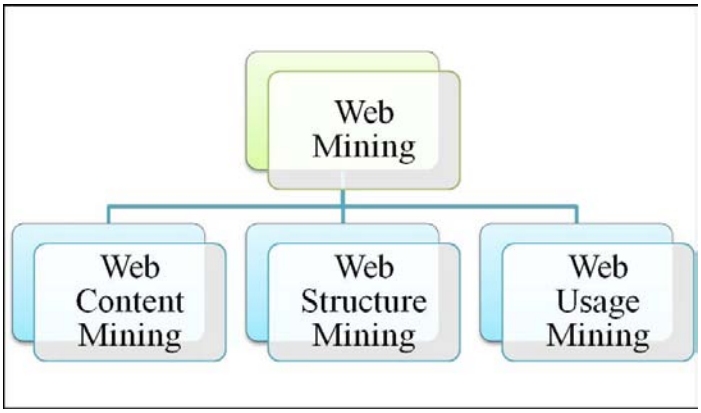

Figure. 2: Web Mining Types

a) ***Web Content Mining:*** The aim of web content mining is to analyze the web content (such as text, multimedia data) that may be located within the same web pages or linked across web pages. Web content mining helps a researcher to understand the content of web pages, provide keyword-based page indexing, web page relevance, web page ranking, web page content summaries, and information related to web search and analysis.

b) ***Web Structure Mining:*** Web structure mining is the process of using graph and network mining theories to comprehend the nodes and hyperlink structures on the Web. It can mine the document structure within a web page or across the different web pages.

c) ***Web Usage Mining:*** Web usage mining focuses on the extraction of useful information from server logs. It tries to discover the patterns that are related to some general or a particular group of users, understand user search patterns and envisage what users are looking for on the Internet [9].

IV. WEB USAGE MINING PROCESS

A detailed web usage mining process with its sub phases is given in figure 3. The three steps involved in Web usage mining process are as follows:

a) ***Data Preprocessing-*** It performs a series of steps covering:

a) Data Cleaning
b) User Identification
c) Session Identification
d) Path Completion
e) Transaction Identification

b) ***Pattern Discovery-*** It involves application of various data mining techniques to processed data like

a) Statistical Analysis
b) Clustering
c) Pattern Matching

c) ***Pattern Analysis-*** It filters out the irrelevant patterns from the identified patterns generated in pattern discovery phase[11].

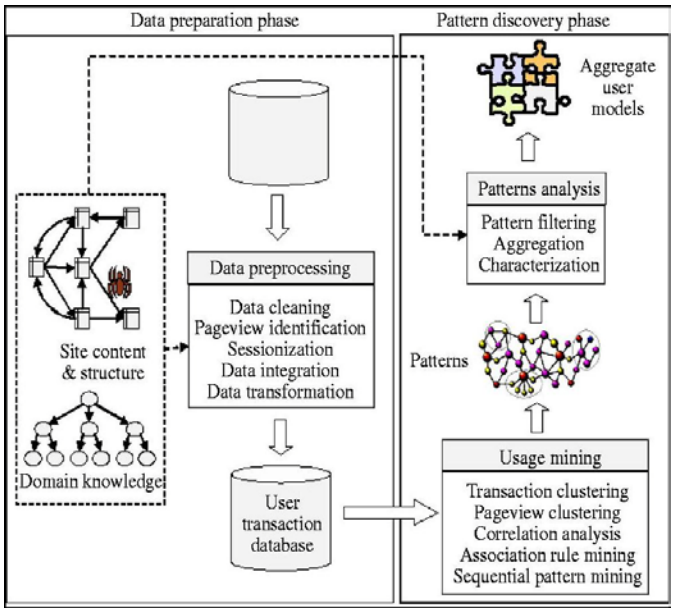

Figure. 3: Detailed Web Usage Mining Process [6]

V. IMPROVISED ALGORITHM OF WEB MINING

The main focus of the purposed solution is on cleaning the raw web log files and inserting the processed data into a relational database so that it is becomes appropriate to apply the mining techniques in the later phases.

Outline of the improvised solution is:

a. Extract the web log and separate its data fields.
b. Store the separated fields in relational database table.
c. Remove irrelevant data by applying data cleaning algorithm on the table where data of log file is stored after separation.

A. Data Field Extraction :

A server log file consists of various data fields (such as IP address, status code, method, path, etc) must be separated out before applying cleaning procedure. This process of

separating out different data fields from single server log entry is identified as data field extraction.

A server uses different characters such as a comma or a space character which works as a separator for separating the fields in a single log entry. The algorithm proposed here for data field extraction uses the space character as a separator to separate the fields of the log file. The log file used for the experimentation is in CLF (Common Log Format) form which means it consists of ten fields, namely,

a) IP address
b) User ID
c) Hostname
d) Timestamp
e) GMT offset
f) Method
g) Path

```
129.173.67.107 - - [23/Feb/2004:14:22:01 -
0500] "GET
/~ai04/_derived/sponsors.htm_cmp_glacier11
0_vbtn.gif HTTP/1.1" 304 0

29.173.67.107 - - [23/Feb/2004:14:22:05 -
0500] "GET /~ai04/submit/ HTTP/1.1" 304 0

129.173.67.107 - - [23/Feb/2004:14:22:05 -
0500] "GET /incoming/cyberstyle.css
HTTP/1.1" 404 2231

137.207.216.174 - - [23/Feb/2004:14:22:10 -
0500] "GET
/~janyst/chat/chatAppletXML.php?id=20
HTTP/1.1" 200 34
```

Figure. 4: Sample Log File

The implementation of the algorithm is done in Java programming language. It makes use of some of Java's inbuilt classes and methods. It is assumed that space character is acting as the separator. The log file is read character by character up to the end and then by using the methods of String Tokenizer class the data fields are broken into tokens and saved in an array.

Algorithm 1:

Input: Log File

Output: Field Separated log fields

Step 1: Initialize token: = null /* token is variable that will contain the items read from the log file */

Step 2: Initialize retTokArr: = null/* retTokArr is an array that will contain the items read from the log file after separation */

Step 3: Find location of the log file to be read.

Step 4: Open file for reading.

Step 5: token: = readFile () /* Read items from the log file character by character in the form of string */

Step 6: while (token! = null) /* Run a while loop until all the items are not read from the file*/

{

retTokArr := token.split (" ") /* Use space as a delimiter to separate out the fields */

}

Step 7: Close the file

Step 8: End

B. Data Storage

The second step describes the storage of field extracted from the log file in a table. Before data storage table named as log table is created in which each entry from the original log file is stored. The sample SQL query for creating the log table with the column names and data types is shown in figure 5.

```
CREATE TABLE LOGTABLE
(
        IPADDRESS VARCHAR2 (50),
        HOSTNAME VARCHAR2 (50),
        USER_NAME VARCHAR2 (50),
        TIME_STAMP VARCHAR2 (50),
        OFFSET VARCHAR2 (50),
        METHOD VARCHAR2 (50),
        PATH VARCHAR2 (250),
        PROTOCOL VARCHAR2 (50),
        STATUS VARCHAR2 (50),
        BYTES VARCHAR2 (50)
)
```

Figure. 5: SQL Query for Table Creation

a. Implementation:

The below code shows the code used to export the data from excel and store into database. MySQL is used as database to store the data and Java is used as front end to implement the changes.

Code:

```
package com.thesis;
import java.io.BufferedInputStream;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Iterator;
import org.apache.poi.hssf.usermodel.HSSFRow;
import org.apache.poi.hssf.usermodel.HSSFSheet;
import org.apache.poi.hssf.usermodel.HSSFWorkbook;
import org.apache.poi.poifs.filesystem.POIFSFileSystem;
import com.db.utils.DbUtils;
public class ReadExcel
{
public static void main(String[] args) throws SQLException
{
try
{
String filename =
"C:\\Users\\Rachit\\workspace\\Thesis\\access_log1.xls";
InputStream input = new BufferedInputStream(new
                FileInputStream(filename));
POIFSFileSystem fs = new POIFSFileSystem(input);
HSSFWorkbook wb = new HSSFWorkbook(fs);
HSSFSheet sheet = wb.getSheetAt(0);
Connection con = DbUtils.getCon();
PreparedStatement pstm = null;
Iterator rows = sheet.rowIterator();
while (rows.hasNext())
{
HSSFRow row = (HSSFRow) rows.next();
System.out.println("\n");
```

```
String ipaddress, hostname, user_name, time_stamp, offset,
method, path, protocol;
int status;
int bytes;
ipaddress = row.getCell(0).getStringCellValue();
hostname = row.getCell(1).getStringCellValue();
user_name = row.getCell(2).getStringCellValue();
time_stamp = row.getCell(3).getStringCellValue();
offset = row.getCell(4).getStringCellValue();
method = row.getCell(5).getStringCellValue();
path = row.getCell(6).getStringCellValue();
protocol = row.getCell(7).getStringCellValue();
status = (int) row.getCell(8).getNumericCellValue();
//bytes= (int)row.getCell(9).getNumericCellValue();
String sql = "INSERT INTO logtable VALUES('" +
ipaddress + "','" + hostname + "','" + user_name + "','"
+ time_stamp + "','" + offset + "','" + method + "','"
+ path + "','" + protocol + "','" + status + "','" + 1+ "')";
pstm = (PreparedStatement) con.prepareStatement(sql);
pstm.execute();
}
pstm.close();
con.close();
System.out.println("Successfully inserted records from excel
to mysql table");
}
catch (IOException ex) {
ex.printStackTrace();
}}}
package com.db.utils;
import java.sql.Connection;
import java.sql.DriverManager;
public class DbUtils
{
static Connection con;
static
{
try
{
Class.forName("com.mysql.jdbc.Driver");
con = (Connection)
DriverManager.getConnection("jdbc:mysql://localhost/thesi
sdb","root","qwerty"); }
catch(Exception e)
{
e.printStackTrace();
}}
public static Connection getCon()
{
return con;
}}
```

C. Data Cleaning:

The third step shows the data cleaning. This retains only those data entries in the log file whose status code is 200, method is GET and file type is except from gif, jpg and css, .txt.

The will removes the gif, jpg, css entries and cleans the web log file.

a. Implementation:

SQL queries are used to perform the data cleaning. First query is used to insert the filterer and clean data into database.

Insert into logtable_updated (ipaddress, hostname, user_name, time_stamp, offset, method, path, protocol, status, bytes) // To insert updated entries into log table.

Below query select the records that have status as 200 and method having GET.

Select ipaddress, hostname, user_name, time_stamp, offset, method, path, protocol, status, bytes from logtable where status= '200' and method='"GET';

VI. RESULT AND ANALYSIS

This section summarizes the result and the analysis of the proposed algorithm. The proposed algorithm performs the cleaning of web log file. The algorithm removes irrelevant records such as records with gif, jpg, css, and txt as suffixes as these have no importance from the perspective of the web usage mining.

Initially, 601 requests were made to the server. There were a total of 36 visitors who contacted the server but only 31 unique IP addresses are identified. The log file consists of 61 entries for image files. It means out of total requests 61 requests are made for the image files. Out of these 61 requests 48 request are made for gif files while only 13 requests are made for jpg files. There were 43 requests that were failed. This is due to the reason that resource was not found. This was the case before the application of clean in algorithm. After the experimentation only 150 useful entries were left. Moreover, the number of unique IP addresses decreases to 16.

Figure 6 shows a graphical representation of above comparison. A bar graph is used to represent the comparison. The blue bar shows the number of accesses before cleaning while the red bar shows the number of accesses after cleaning. The bar chart is clearly showing that there is a major decrease in the number of request.

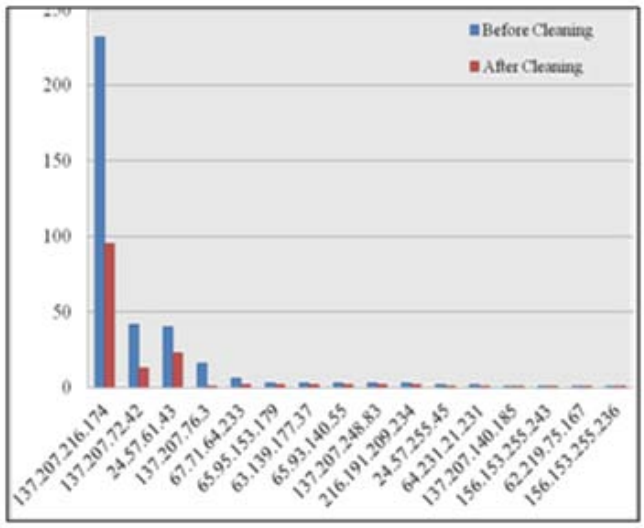

Figure 6: Bar Chart Showing Comparison in Number of Access

Due to reduction in the number of accesses by the unique IP addresses, the size of the log also decreases.

The graph makes it clear that there is a severe change in both the size and number of records after data cleaning. From this observation if calculate the percentage amount of decrease in the size of log file, it gives a reduction of 74% (see table 6) which is quite a major value.

Table 1: Results of Data Cleaning

Web server Log File	Result
Original Size	60 KB
Reduced Size	15.6 KB
Percentage in Reduction	74.00

VII. CONCLUSION

a. The web cleaning step of data preprocessing is crucial as the result of this step have an impact on the accuracy of results of the later phases.
b. An improvised technique for performing the data cleaning technique on server log was proposed.
c. The proposed approach showed a quite salient reduction in the number of records and in the log files size and hence increases the quality of the available data.

VIII. FUTURE SCOPE

a. The research presented in this paper is in an emerging stage. However, the subjective interpretation of the technique is very ingenious and can propose a lot of scope to be extended on to other problem domains.
b. The research can be extended to the log file of other formats such as extended common log format which consists of more fields than a common log format.
c. Many problems such as applications of user identification, session identification, and path completion are not discussed.

IX. REFERENCES

[1]. Frawley W.J., Piatetsky-Shapiro G. and Matheus C.J., "Knowledge Discovery in Databases: An Overview", AI Magazine, vol. 13, no. 3, pp. 57-70, 1992.

[2]. Fayyad U., Piatetsky-Shapiro G. and Smyth P., "From Data Mining to Knowledge Discovery in Databases", AI Magazine, vol. 17, no. 3, pp. 37-54, 1996.

[3]. Srivastava J., and Cooley R., "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data", ACM SIGKDD Explorations, vol. 1, no. 2, pp. 12-23, January 2000.

[4]. Bharat K. and Broder A., "A Technique for Measuring the Relative Size and Overlap of Public Web Search Engines", in Proceedings of the 7th World-Wide Web Conference, pp. 379-388, 1998.

[5]. Singh B., Singh H.K., "Web Data Mining Research", in Proceedings of 2010 IEEE International Conference on Computational Intelligence and Computing Research, pp. 1-10, December 2010.

[6]. Bayir M.A., Toroslu I.H., Cosar A. and Fidan G.,"Smart Miner: A New Framework for Mining Large Scale Web Usage Data", in Proceedings of the 18th International Conference on World Wide Web, pp. 161-170, 2009.

[7]. Cooley R., "Web Usage Mining: Discovery and Application of Interesting Patterns from Web data", PhD thesis, University of Minnesota, Dept. of Computer Science, May 2000.

[8]. Singh B., Singh H.K., "Web Data Mining Research", in Proceedings of 2010 IEEE International Conference on Computational Intelligence and Computing Research, pp. 1-10, December 2010.

[9]. Zhang Q., and Segall R. S., "Web Mining: A Survey of Current Research, Techniques, and Software", International Journal of Information Technology & Decision Making, vol. 7, no. 4, pp. 683-720,2008.

[10]. Borges J. and Levene M., "Data Mining of User Navigation Patterns", in Proceedings of the WEBKDD'99 Workshop on Web Usage Analysis and User Profiling, pp. 31-39, August 1999.

[11]. Madria S.K., Bhowmick S.S., Ng W.K., and Lim E.P., "Research Issues in Web data Mining", in Proceedings of First International Conference Data Warehousing and Knowledge Discovery, pp. 303-312, 1999.

[12]. Etzioni O., "The World Wide Web: Quagmire or Gold Mining?", Communications of the ACM, vol. 39, no. 11, pp. 65-68, November 1996.

[13]. Blockeel H. and Kosala R., "Web Mining Research: A Survey", ACM SIGKDD Explorations, vol. 2, no. 1, pp. 1-15, June 2000.