ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Special Issue 1, March 2017

Eclat Algorithm on Web Log Data for Mining the Frequent Link

M. Sathya, Dr. P. Isakki @ Devi

Research Scholar, Department of Computer Science, Ayya Nadar Janaki Ammal College, Sivakasi, Tamil Nadu, India

Assistant Professor, Department of Computer Applications, Ayya Nadar Janaki Ammal College, Sivakasi, Tamil Nadu, India

ABSTRACT**:** Web Usage Mining is one of the parts of web mining and extracts the web users' behavior from web log file. The proposed system consists of three phases, namely data preprocessing, pattern discovery, and pattern analysis. Raw web log data may contain noise and impurities. By using some data preprocessing techniques that noise will be removed. Data preprocessing phase is the most important one because it makes the data with good quality. In Pattern discovery phase, the users' navigational pattern and rules are extracted by using association rule algorithm. Pattern analysis phase is to analyze and visualize the rules. In this paper, first preprocessing can be done with data cleaning, user identification, and session identification. The objective of this research paper is to identify the frequent link from web log data by using the Eclat algorithm.

KEYWORDS: Web usage mining; Data cleaning; User Identification; Session Identification; Eclat Algorithm;

I. INTRODUCTION

Web mining is one of the applications of data mining techniques to extract knowledge from web log data, including web documents, hyperlinks between documents, usage logs of web sites, etc. website is a collection of web pages that is document accessible through the World Wide Web on the internet. Web mining is a combination of data mining and World Wide Web. It consists of types, namely web content mining, web structure mining and web usage mining. Web content mining is used to extract useful information from the contents of web documents. Content data is the collection of facts in a designed web page. It may consist of text, images, audio, video, lists and tables. Web structure mining is the process of discovering structure information from the web. Web structure mining is used to analyze the node and connection structure of a web site. It can be divided into two kinds that are hyperlinked and document structure [17].

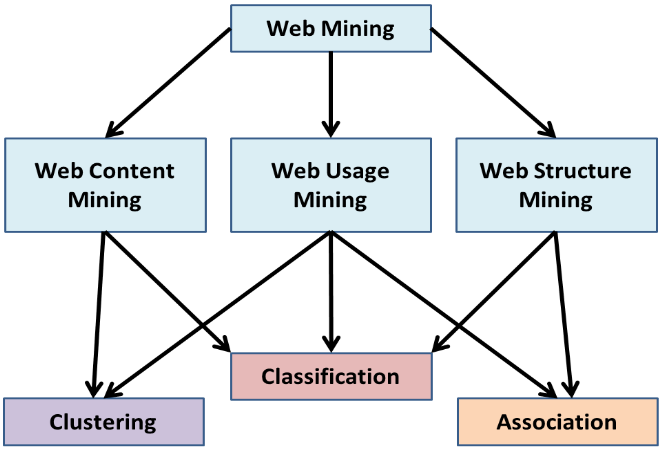


Fig. 1. Framework of Web mining

Web usage mining is used to discover interesting usage patterns from web data, in order to understand the needs of web-based applications. It is the third type of web mining and also the application of data mining techniques. The web content and structure mining mines the primary data on the web but web usage mining utilize the secondary data derived from the interactions of the users. Web usage mining analysis results of user interactions with a web server, including weblogs, click streams, and database transactions at a web site of a group of related sites [2].

Web Usage Mining consists of a three phase process

a) Pre-processing / Data Preparation
b) Pattern Discovery
c) Pattern Analysis

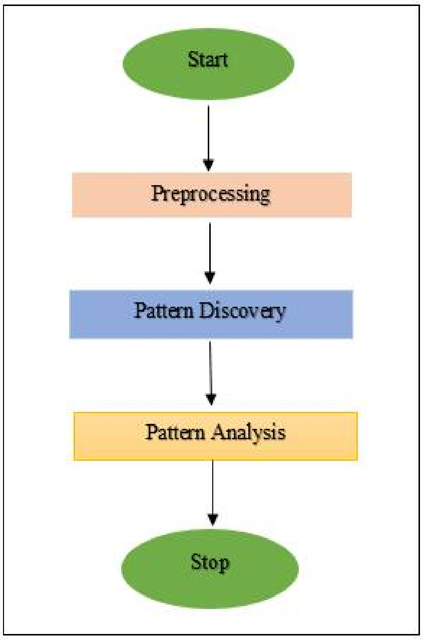


Fig. 2. The Three Phases of web usage mining

A. PREPROCESSING

Data preprocessing is the essential process and must be performed prior to applying data mining algorithms to the data sources. The aim of data preprocessing is identifying the unique users, user sessions and transactions are presented in this paper.

B. PATTERN DISCOVERY

The second phase of web usage mining is pattern discovery. In this phase, patterns are discovered from preprocessed data by using some data mining methods like association, clustering and statistical analysis and so on.

C. PATTERN ANALYSIS

This is the last phase in the Web Usage Mining process. In this phase patterns are analyzed to extract the useful information from result of pattern discovery data by using knowledge query mechanisms such as SQL or data cubes to perform OLAP operations.

II. LITERATURE REVIEW

A literature review discusses published information in a particular subject area, and sometimes information in a particular subject area within a certain time period [16].

A literature review can be just a simple summary of the sources, but it usually has an organizational pattern and combines both summary and synthesis. A summary is a recap of the important information about the source, but a synthesis is a re-organization, or a reshuffling, of that information. It might give a new interpretation of old material or combine new with old interpretations. Due to the dependent situation, the literature review may evaluate the sources and advise the reader on the most pertinent or relevant [16].

Rathi, A., *et al.* [1] presented this paper to understand the web usage mining process such as preprocessing of web usage data and also the finding of frequent Patterns and their analysis. And also the comparison of both algorithms on the same dataset is done. Due to more use of the internet, the log files are increasing at a higher rate according to size. The Preprocessing plays an important role in the efficient mining process because data in Log files are normally noisy and not distinct.

Sriram, R., *et al.* *[2]* proposed an efficient pre-processing technique and an innovative Hashing technique **-** (a Hash table and a Hash function have been proposed) to identify a Distinct User for Web Usage Mining. The proposed pre-processing technique has been evaluated by comparing with existing pre-processing techniques to prove its accuracy and efficiency. Similar the Hashing techniques have been compared with existing search methodologies and it has been proved that the proposed technique is quick in searching according to Big O notation.

Ma, Z., *et al.* *[9]* proposed an improved Eclat algorithm called Eclat_growth which is based on the increased search strategy. There are three main steps in the Eclat_growth algorithm. First, it scans the database and stores it into a table using vertical data format. Then, it builds an increased two-dimensional pattern tree and the TID_sets of itemsets in the vertical data format table are added into the pattern tree row by row. New frequent itemsets are generated by combining the new added item data with the existing frequent itemsets in the pattern tree. Finally, all frequent itemsets can be found by picking up all nodes of the pattern tree. In the process of generating new, frequent itemsets, the prior knowledge is used to full clip the candidate itemsets. In the process of generating an intersection of two itemsets and calculating the support degree, we proposed a new method called BSRI (Boolean array setting and retrieval by indexes of transactions) to reduce the run time. By comparing Eclat_growth with Eclat, Eclat-diffsets, Eclat-opt and hEclat, it is indicated that Eclat_growth has the highest performance in mining associating rules from various databases.

Vijayarani, S., *et al.* *[5]* described that mining frequent items in data streams using Eclat and dynamic itemset mining algorithms and finding the performance and drawbacks of these two algorithms. Most commonly used traditional association rule mining algorithms are Apriori algorithms, Partitioning algorithms, Pincer-Search algorithms, FPGrowth algorithms and dynamic item set mining algorithms, Eclat algorithms and so on. The performance factors used are number of frequent items generated using different thresholds and execution time. From the experimental results we come to know that the performance of Eclat algorithm is better than the dynamic item set counting algorithm. Keywords - Data streams, Association rules, Frequent Items, Éclat algorithm, Dynamic Item Set Counting Algorithms.

Yu, X., *et al.* *[12]* compared with a traditional Eclat algorithm, the results of experiments show that the Bi-Eclat algorithm gains better performance on several public databases given. Furthermore, the Bi-Eclat algorithm is applied in analyzing combination principles of prescriptions for Hepatitis B in Traditional Chinese Medicine, which shows its efficiency and effectiveness in practical usefulness.

III. DATA PREPROCESSING

Data preprocessing is required stage, which can done by data collection, data cleaning, user identification, and session identification. The goal of preprocessing is to improve the quality and accuracy of data.

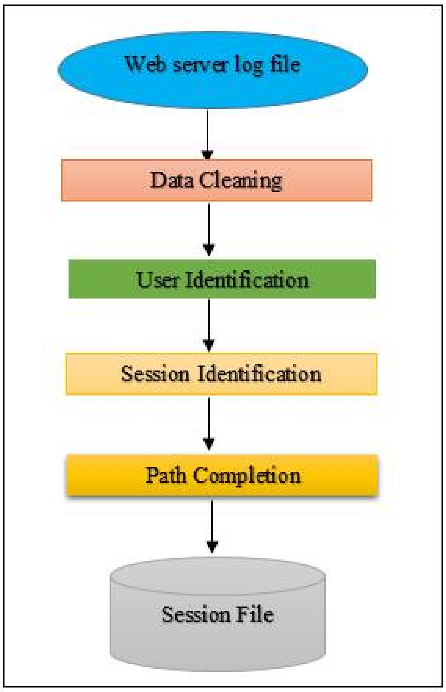


Fig. 3. Data Preprocessing Techniques

A. *DATA COLLECTION*

The web log data have been collected from one educational institute's website. Initially the data size is 1000 records X 4 attributes.

B. *DATA CLEANING*

The data cleaning is an important process of data preprocessing. It is used to remove the irrelevant and duplicate records from the log file that are not required for mining. After cleaning is done, the data set will be reduced from 1000 to 910. In that only 20 large records are used for further processing.

C. *USER IDENTIFICATION BY IP ADDRESS AND DATE*

User identification means identifying individual users by using their IP address. There are following rules to identify unique users:

1) If there is new IP address then there is a new user;

2) If the IP Address is same but the visited date is different then there is also a new user. In this paper, 149 unique users are identified. From that only 20 Users web logs details are used for mining frequent link.

D. *SESSION IDENTIFICATION*

Session means the duration of the user's spent on a web page. Session identification is used to divide the page access of each user into different sessions. There are many sessions for the same users are possible. Two methods are available for identifying or creating the sessions of the user as such depends on time and depends on navigation. In this research session ids are created by using depends on time, which is calculated by the difference between two time stamps of the same user.

IV. PROPOSED WORK

A. *ECLAT ALGORITHM FOR WEB MINING*

The Eclat algorithm is used to perform itemset mining. Itemset mining means to find frequent patterns in web log data. This algorithm has three traversal approaches like top-down, bottom-up and hybrid. Each item is stored together with its tidlist and compute the support of an itemset by using the intersection based approach. If itemsets are small in number it requires less space than a apriori. Eclat algorithm is based on two main steps:

Step1. Candidate generation
Step2. Pruning

In step 1, each k-*itemset* candidate is generated from two frequent$(k-1)-itemsets$ and then its support is counted, if its support count is lower than the threshold, then it will be removed, otherwise it is frequent itemsets and used to generate$(k+1)-itemsets$.

B. *ECLAT ALGORITHM IN PSEUDOCODE*

```
Input: $E((i_1, t_1), \ldots (i_n, t_n))|P), s_{min}$
Output: $F(E, s_{min})$
for all $i_j$ occuring in E do
    $P := P \cup i_j$ // add $i_j$to create a new prefix
    init($E'$) // initialize a new equivalence class with the new prefix P
    for all $i_k$ occuring in E such that $k > j$ do
        $t_{tmp} = t_j \cap t_k$
        if $|t_{tmp}| \geq s_{min}$ then
            $E' := E \cup (i_k, t_{tmp})$
            $F = F \cup (i_k \cup P)$
        end if
    end for
    if $E' \neq \{\}$ then
        Eclat($E', s_{min}$)
    end if
end for
```

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

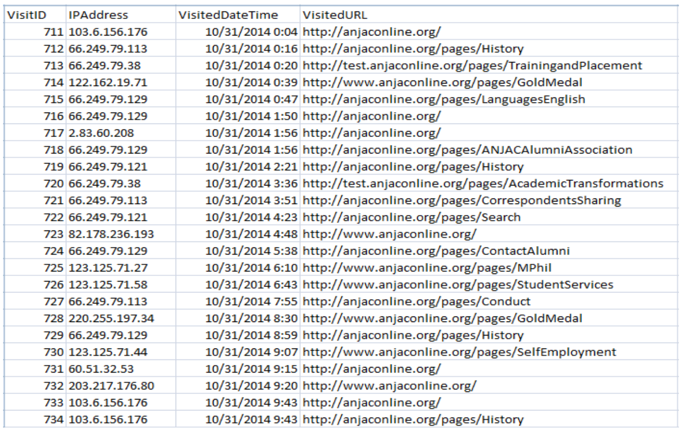
International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

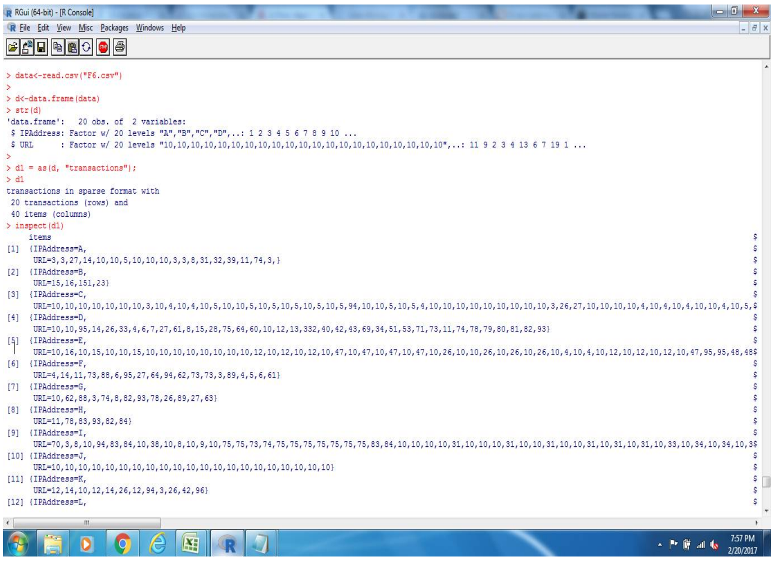
Vol. 5, Special Issue 1, March 2017

V. EXPERIMENTAL RESULTS IN FIGURES



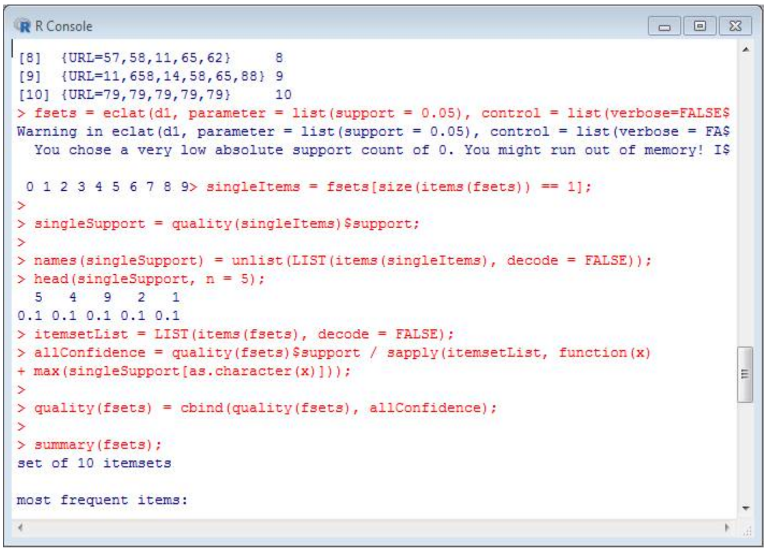
VisitID	IPAddress	VisitedDateTime	VisitedURL
711	103.6.156.176	10/31/2014 0:04	http://anjaconline.org/
712	66.249.79.113	10/31/2014 0:16	http://anjaconline.org/pages/History
713	66.249.79.38	10/31/2014 0:20	http://test.anjaconline.org/pages/TrainingandPlacement
714	122.162.19.71	10/31/2014 0:39	http://www.anjaconline.org/pages/GoldMedal
715	66.249.79.129	10/31/2014 0:47	http://anjaconline.org/pages/LanguagesEnglish
716	66.249.79.129	10/31/2014 1:50	http://anjaconline.org/
717	2.83.60.208	10/31/2014 1:56	http://anjaconline.org/
718	66.249.79.129	10/31/2014 1:56	http://anjaconline.org/pages/ANJACAlumniAssociation
719	66.249.79.121	10/31/2014 2:21	http://anjaconline.org/pages/History
720	66.249.79.38	10/31/2014 3:36	http://test.anjaconline.org/pages/AcademicTransformations
721	66.249.79.113	10/31/2014 3:51	http://anjaconline.org/pages/CorrespondentsSharing
722	66.249.79.121	10/31/2014 4:23	http://anjaconline.org/pages/Search
723	82.178.236.193	10/31/2014 4:48	http://www.anjaconline.org/
724	66.249.79.129	10/31/2014 5:38	http://anjaconline.org/pages/ContactAlumni
725	123.125.71.27	10/31/2014 6:10	http://www.anjaconline.org/pages/MPhil
726	123.125.71.58	10/31/2014 6:43	http://www.anjaconline.org/pages/StudentServices
727	66.249.79.113	10/31/2014 7:55	http://anjaconline.org/pages/Conduct
728	220.255.197.34	10/31/2014 8:30	http://www.anjaconline.org/pages/GoldMedal
729	66.249.79.129	10/31/2014 8:59	http://anjaconline.org/pages/History
730	123.125.71.44	10/31/2014 9:07	http://www.anjaconline.org/pages/SelfEmployment
731	60.51.32.53	10/31/2014 9:15	http://anjaconline.org/
732	203.217.176.80	10/31/2014 9:20	http://www.anjaconline.org/
733	103.6.156.176	10/31/2014 9:43	http://anjaconline.org/
734	103.6.156.176	10/31/2014 9:43	http://anjaconline.org/pages/History

Fig. 4. Shows the web log data it has id, IP address, visitdatetime, url.



```
> data<-read.csv("F6.csv")
>
> d<-data.frame(data)
> str(d)
'data.frame':   20 obs. of  2 variables:
 $ IPAddress: Factor w/ 20 levels "A","B","C","D",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ URL      : Factor w/ 20 levels "10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10",..: 11 9 2 3 4 13 6 7 19 1 ...
>
> d1 = as(d, "transactions");
> d1
transactions in sparse format with
 20 transactions (rows) and
 40 items (columns)
> inspect(d1)
     items                                                                                                                                  $
[1]  {IPAddress=A,                                                                                                                          $
      URL=3,3,27,14,10,10,5,10,10,10,3,3,8,31,32,39,11,74,3,}                                                                               $
[2]  {IPAddress=B,                                                                                                                          $
      URL=15,16,151,23}                                                                                                                     $
[3]  {IPAddress=C,                                                                                                                          $
      URL=10,10,10,10,10,10,10,3,10,4,10,4,10,5,10,10,5,10,5,10,5,10,5,10,5,94,10,10,5,10,5,4,10,10,10,10,10,10,10,10,10,3,26,27,10,10,10,10,4,10,4,10,4,10,10,4,10,5,$
[4]  {IPAddress=D,                                                                                                                          $
      URL=10,10,95,14,26,33,4,6,7,27,61,8,15,28,75,64,60,10,12,13,332,40,42,43,69,34,51,53,71,73,11,74,78,79,80,81,82,93}                   $
[5]  {IPAddress=E,                                                                                                                          $
      URL=10,16,10,15,10,10,15,10,10,10,10,10,10,10,10,12,10,12,10,12,10,47,10,47,10,47,10,47,10,26,10,10,26,10,26,10,26,10,4,10,4,10,12,10,12,10,12,10,47,95,95,48,48$
[6]  {IPAddress=F,                                                                                                                          $
      URL=4,14,11,73,88,6,95,27,64,94,62,73,73,3,89,4,5,6,61}                                                                               $
[7]  {IPAddress=G,                                                                                                                          $
      URL=10,62,88,3,74,8,82,93,78,26,89,27,63}                                                                                             $
[8]  {IPAddress=H,                                                                                                                          $
      URL=11,78,83,93,82,84}                                                                                                                $
[9]  {IPAddress=I,                                                                                                                          $
      URL=70,3,8,10,94,83,84,10,38,10,8,10,9,10,75,75,73,74,75,75,75,75,75,75,75,83,84,10,10,10,10,31,10,10,10,31,10,10,31,10,10,31,10,31,10,31,10,33,10,34,10,34,10,3$
[10] {IPAddress=J,                                                                                                                          $
      URL=10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10}                                                                   $
[11] {IPAddress=K,                                                                                                                          $
      URL=12,14,10,12,14,26,12,94,3,26,42,96}                                                                                               $
[12] {IPAddress=L,                                                                                                                          $
```

Fig. 5. Dataset converts into a transaction.

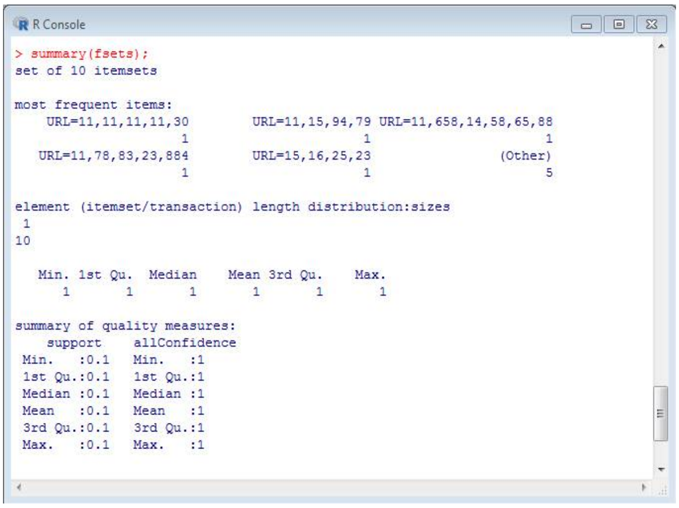


```
[8]  {URL=57,58,11,65,62}     8
[9]  {URL=11,658,14,58,65,88} 9
[10] {URL=79,79,79,79,79}     10
> fsets = eclat(d1, parameter = list(support = 0.05), control = list(verbose=FALSE$
Warning in eclat(d1, parameter = list(support = 0.05), control = list(verbose = FA$
  You chose a very low absolute support count of 0. You might run out of memory! I$

 0 1 2 3 4 5 6 7 8 9> singleItems = fsets[size(items(fsets)) == 1];
>
> singleSupport = quality(singleItems)$support;
>
> names(singleSupport) = unlist(LIST(items(singleItems), decode = FALSE));
> head(singleSupport, n = 5);
  5   4   9   2   1
0.1 0.1 0.1 0.1 0.1
> itemsetList = LIST(items(fsets), decode = FALSE);
> allConfidence = quality(fsets)$support / sapply(itemsetList, function(x)
+ max(singleSupport[as.character(x)]));
>
> quality(fsets) = cbind(quality(fsets), allConfidence);
>
> summary(fsets);
set of 10 itemsets

most frequent items:
```

Fig. 6. Eclat algorithm is applied in training web log data set



```
> summary(fsets);
set of 10 itemsets

most frequent items:
    URL=11,11,11,11,30          URL=11,15,94,79 URL=11,658,14,58,65,88
                     1                        1                      1
   URL=11,78,83,23,884          URL=15,16,25,23                (Other)
                     1                        1                      5

element (itemset/transaction) length distribution:sizes
 1
10

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      1       1       1       1       1       1

summary of quality measures:
    support     allConfidence
 Min.   :0.1   Min.   :1
 1st Qu.:0.1   1st Qu.:1
 Median :0.1   Median :1
 Mean   :0.1   Mean   :1
 3rd Qu.:0.1   3rd Qu.:1
 Max.   :0.1   Max.   :1
```

Fig. 7. It displays the most frequent items.

The goal of the paper is to mine the frequent link from the web log file by using the Eclat algorithm. Frequent link mining also used to find information like set of pages repeatedly accessed together by web users. The administrator can modify the website according to the mining result.

VI. Conclusion

Web Usage Mining is one of the parts of web mining and extracts the web users' behavior from log files**.** Preprocessing is done by data cleansing, user identification, session identification. Data cleaning is used to reduce the size of web log file and also improves the quality of contents in the log file. Then the Eclat algorithm is applied in preprocessed data for mining the frequent link from web log data. The Eclat algorithm has high performance compared to Apriori.

REFERENCES

[1] Rathi, A., and Raipurkar, A., "Approach for processing of Web Usage Data." International Research Journal of Engineering and Technology(IRJET), 2016, Vol.3, Issue.07, pp: 339-343.
[2] Sriram, R., and Malliga, R., "Innovative Pre-Processing Technique and Efficient Unique User Identification Algorithm for Web Usage Mining." International Journal of Advanced Research in Computer Science and Software Engineering, 2016, Vol.6, Issue.02, pp: 85-91.
[3] Umarani, J., and Karpagam, K., "Investigation of User Identification Methods in Pre-Processing Phase of Web Usage Mining." International Journal of Engineering Science and Computing, 2016, Vol.6, Issue.08, pp: 2954-2956.
[4] Kumari, A, G, K., and Shetty, S., "Web Usage Mining: Web Log Pre-Processing and Online Visitor's Frequent Pattern Discovery." International journal of Innovative Research in Computer and Communication Engineering, 2016, Vol.4, Issue.04, pp: 5192-5199.
[5] Patil, S, S., and Khandagale, H, P., "Enhancing Web Navigation Usability Using Web Usage Mining Techniques." International research Journal of Engineering and Technology(IRJET), 2016, Vol.4, Issue.06, pp: 2828-2834.
[6] Kaur, M., and Gurm, K, P., "Web Usage Mining Through FP Split and Apriori Algorithm." International Journal of Technology and Computing(IJTC), 2016, Vol.2, Issue.9, pp: 436-440.
[7] Agrawal, N., and Jawdekar, A., "A Survey Report On Current Research and Development of Data Processing In Web Usage Data Mining." International Journal of Database Theory and Application, 2016, Vol.9, Issue.5, pp: 101-110.
[8] Deotale, K., and Honale, S., "Domain Specific Approach For Weblog Mining." International Journal of Engineering Sciences & Research Technology, 2016, Vol.5, Issue.3, pp: 406-414.
[9] Ma, Z., Yang, J., Zhang, T., and Liu, F., "An Improved Eclat Algorithm for Mining Association Rules Based on Increased Search Strategy." International Journal of Database Theory and Application, 2016, Vol. 9, Issue.5, pp: 251-256.
[10] Vijayarani, S., and Prasannalakshmi, R., "Frequent Items Mining in Data Streams." International Journal of Research in Applied Science & Engineering Technology(IJRASET), 2015, Vol. 3, Issue.4, pp: 358-366.
[11] Yu, X., and Wang, H., "Improvement of Eclat Algorithm Based on Support in Frequent Itemset M ining." Journal of Computers, 2014, Vol. 9, Issue. 9, pp: 2116-2123.
[12] Kumar, A., Singh, O., Rishiwal, V., Dwivedi, K, R., and Kumar, R., "Association Rule Mining On Web Logs For Extracting interesting Patterns Through WEKA Tool." International Journal of Advanced Technology in Engineering and Science, 2015, Vol.3, Issue.1, pp: 134-140.
[13] Kumar, V, S., Kumaresan, S, A., and Jayalakshmi, U., "Frequent Pattern Mining in Web Log Data Using Apriori Algorithm." International Journal of Emerging Engineering Research and Technology, 2015, Vol.3, Issue.10, pp: 50-55.
[14] Parekh, A, M., Patel, A, S., Parmar, S, J., and Patel, V, R., "Web Usage Mining: Frequent Pattern Generation Using Association Rule Mining and Clustering." International Journal of Engineering Research & Technology(IJERT), 2015, Vol.4, Issue.4, pp: 1243-1246.
[15] Deghaidy, M, M., Badran, M, K., and Mohamed, I, G., "Web Recommendation Framework based on Association Rules Coverage to be Applied for Site Modification." International Journal of Computer Applications, 2014, Vol.91, Issue.2, pp: 28-33.
[16] http://writingcenter.unc.edu/handouts/literature-reviews/
[17] http://dmr.cs.umn.edu/Papers/P2004_4.pdf