# A Framework for Clustering & Enhanced Approach for Frequent Patterns in Web Usage Mining

A. A. Abd El-Aziz [1], P. Senthil Pandian[2], Saleh N Almuayqil[3] , Amer S Alruwaili[4]

*[1,3]Assistant Professor, Department of Information Systems, College of Computer & Information Sciences, Jouf University, Saudi Arabia*

*[4]College of Computer & Information Sciences, Jouf University, Saudi Arabia*

*[2]Assistant Professor, Department of Computer Science and Engineering, P.T.R College of Engineering & Technology, Madurai, India*

*E-mail: aa.eldamarany@ju.edu.sa, psenthilpandian@gmail.com, snmuayqil@ju.edu.sa, amer@ju.edu.sa*

## Abstract

*Tremendous measure of data's are accumulated and seen through World Wide Web by various clients. The client rehearses their perspectives by entering hypertext certifications by web with an enormous storehouse of site pages and web utilization digging process is fundamental for effective site the board, personalization, business and bolster administrations, and system traffic stream investigation, and so on., Web page contains pictures, content, recordings and other sight and sound and web log document holds the entrance history clients in the sites. The log document will have some loud and uncertain information which may influence the information mining procedure and enormous amount of web traffic ought to be taken care of adequately to secure wanted data. So the log record ought to be preprocessed to improve the nature of information. Preprocessing comprises of information cleaning and information separating, client ID and session distinguishing proof. Two arrangements of log documents are gathered and prepared to get trial results. This paper shows a structure for client and session preprocessing and bunching with Hidden Damage Data algorithm (HDD) and furthermore dissects the navigational conduct of clients through an Enhanced Conviction Frequent Pattern Mining Algorithm (CFPMA) to distinguish visit designs in web log information. The trial result shows that the proposed strategy accomplishes low execution time and higher exactness when contrasted and the other existing techniques.*

**Keywords:** *web log file, clustering, conviction value, least byte record, frequent patterns*

## 1. Introduction

In the present period web administrations, online data framework and keeping up web log information has gotten significant for the suppliers of web administrations. Web log information comprises of data about the client's web perusing history. The majority of web log information is naturally created by web servers. The data accessible on the World Wide Web has been a touchy development over the most recent couple of decades. To locate the significant data effectively and unequivocally the clients need to have the successful inquiry tools. For example, to decrease the traffic load, the web service providers give the best approach to anticipate the client practices and customize data. By and large web log mining is grouped into web structure mining, web content mining and web usage mining. Web usage mining abuses information mining methods to achieve significant data from the route conduct of the World Wide Web clients. Web structure mining talks about the hyperlink structure of the web. Web structure mining is alluring to organizations including the administration offices to order dangers and battle against psychological oppression. Web content mining is centered on the development of strategies to help clients to discover the web archives. Web use mining is the way toward getting valuable information from server logs. It is the utilization of information mining ways to deal with finds intriguing use designs from the web. To improve site utility and client fulfillment, this paper proposed another philosophy for the procedure of web use mining steps. For extraction of client designs this paper proposed a total preprocessing procedure. For information preprocessing Hidden Damage Data (HDD) algorithm is proposed. Hyper Text Transfer Protocol (HTTP) and Hash set procedures are utilized in HDD. After information cleaning and separating, distinguishing proof of client dependent on IP address and recognizable proof of session dependent on client time, time interim and client session is proposed. The determined sessions are at last grouped utilizing a web session bunching and client group. The two kinds of bunching are utilized for web personalization to infer use profiles. At long last aggregative grouping is created by consolidating client and session bunch. Example extraction process extricates fascinating examples from web logs. Example revelation draws upon algorithms and techniques created from different fields, for example, information mining, measurements, AI and example acknowledgment. The regular example disclosure approaches are applied on crude information right now. Example examination is the last stage in the web utilization mining process. Right now, system is intended to remove aggregative bunching and regular thing sets from the web log information is proposed by utilizing Conviction Frequent Pattern Mining Algorithm (CFPMA). The remainder of the paper is sorted out as follows. Section II exhibits a portrayal about the past research which is pertinent to the examination of preprocessing, bunching procedure and example age. Section 3 includes the definite portrayal about the proposed strategy. Section 4 introduces the exhibition investigation. This paper finishes up in Section 5.

## 2. Related Work

Bianco et al [2] displayed an examination of web usage mining in the site OrOliveSur.com. This paper depicted the arrangement of eliminates conveyed including information preprocessing, information assortment, extraction and examination of information. By utilizing unaided and regulated information mining algorithms through illustrative errands, for example, bunching, affiliation and subgroup revelation, and the information are removed. The outcomes were examined to give a few rules to improving site utility and client fulfillment. In traditional web use mining semantic data about the page content doesn't partake in the example age process. To improve and guarantee the nature of dug models for existing procedure mining approaches, Ly et al [7] built up an information change and preprocessing methods steps. The idea of semantic log cleansing dependent on space explicit imperatives was proposed. The practicality of the methodology was exhibited dependent on a contextual investigation in advanced education space.

Mishra et al [9] proposed the process of web usage mining can be applied in e-learning systems in order to anticipate the marks will obtain in the final exam of a course. A specific model for mining tool was developed to the use of experts in data mining and also for newcomers like instructors and courseware authors. By applying the pattern recognition techniques for web log data, Gupta et al [4] analyzed the web usage mining. Pattern recognition is defined as the act of taking in raw data and making an action based on the category of the pattern. Sudheer Reddy et al [15] analyzed the identification of web usage patterns based on the user’s interest or choice, thereby creating an intelligent semantic-based web usage mining technique.

Taherizadeh et al 16] introduced a procedure to consolidate web content mining into web use mining. To identify valuable data and affiliation runs about clients' practices, the printed substance of website pages is gathered through extraction of incessant word groupings, which are joined with web server log documents. Thakare et al [17] proposed a viable and complete preprocessing of access stream before the genuine mining procedure can be performed. To make significant information source, the log record from various sources experiences distinctive preprocessing stages.

Uma Maheswari et al [18] introduced the algorithms to join the log documents from various servers, clean the fuse web log record, and recognize the clients and to build up the sessions for every client. In web log information, Yu et al [21] proposed web consecutive examples utilizing the hole compelled strategy. By expelling unessential or excess things, gathering the comparative clients and reproducing the web log information into a lot of tuples compelled by visiting time, pre-procedure of the crude web log information was presented. Sheetal Raiyani et al [12] utilized the hole BIDE algorithm in web log information with a less help edge and hole limitations and web got to design which were shut consecutive examples with hole imperatives were created.

The impact of semantic information on the examples produced for web use mining was researched by Sudheer Reddy et al [14]. A structure was created to incorporate the semantic information into web route design age process. Mishra et al [8] displayed the continuous navigational examples comprised of philosophy occurrences rather than website page addresses. An assessment component including page proposal estimated the nature of created designs. The trial results indicated that the utilization of semantic information in route design age improved example quality and produced exact proposals. The weighted regular example mining method was utilized to decide visit designs by thinking about the loads of examples. The weighted backings of examples were coordinated to prune weighted inconsistent examples. Yun et al [20] proposed vigorous idea of mining accurate weighted incessant examples. An effective Hierarchical Frequent Pattern Analysis (HFPA) approach by Sudhamathy et al [13] mined affiliation rules from web logs by using an ordinary Apriori algorithm. The intriguing quality measures were utilized to sort the found affiliation controls after the utilization of pruning technique. The standards that were positioned exceptionally as per the intriguing quality estimates were important to the site head.

Bhattacharya et al [1] displayed the relative examination of Apriori algorithm and frequent pattern algorithm for visit design mining in web log information. Apriori was the least complex algorithm utilized for mining successive examples from the exchange database. However, the primary inconvenience of the Apriori algorithm was that the age of applicant set was exorbitant, especially if there is a presence of countless examples or long examples. Huge thing set property was utilized by the Apriori algorithm, which was anything but difficult to actualize, yet over and again performed database check. Apriori additionally expended more opportunity to filter the enormous continuous examples. Raiyani et al [11] approach could be a genuine site that contained the difficult parts of genuine web usage mining, including outer information portraying the cosmology of the web content.

A powerful system for visit design digging utilizing web logs for web usage mining was clarified by Raju et al [10]. The methodology was known as Intelligent Frequent Pattern Analysis. Right now, technique was applied to mine affiliation rules from web logs by using an ordinary Apriori algorithm, however with barely any changes for improving the intriguing quality of the created rules. Before the affiliation rules were mined, the information was grouped with fluffy bunching, which was then improved through hereditary algorithm. An improved web personalization approach found via Carmona et al [3] on client intrigued indexes. The strategy accomplished lesser memory necessity and better handling time when contrasted and the non-weighted access design mining draws near. Joel et al [5] gave two novel tree structures, gradual Weighted Frequent Pattern (WFP) tree dependent on weight rising request and steady WFP tree dependent on recurrence dropping request. They were effective for intelligent and steady WFP mining to utilize the past mining results and current tree structure when a base help edge was adjusted or a database was refreshed. An equal, disseminated algorithm by Lin et al [6] was utilized to find social recurrence designs from huge datasets. A near investigation of affiliation rule mining algorithms, for example, AIS, SETM and Apriori and the AIS algorithm comprised of two stages. The age of incessant thing sets was acted in the primary stage. The certain and visit affiliation rules were produced in the subsequent stage. Less number of up-and-comer thing sets was delivered for testing in each database pass by the Apriori algorithm.

## 3. Proposed Technology

The flow of the proposed work is shown in Figure 1.

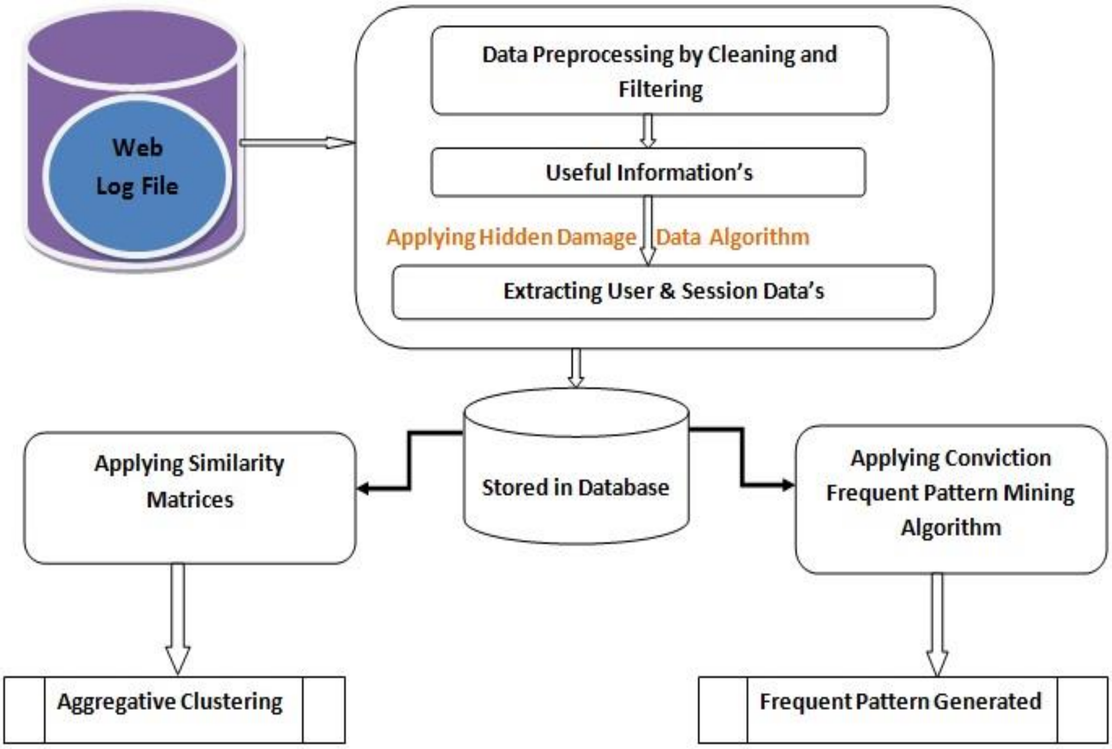

Web
Log File
Data Preprocessing by Cleaning and Filtering
Useful Information's
Applying Hidden Damage Data Algorithm
Extracting User & Session Data's
Stored in Database
Applying Similarity Matrices
Applying Conviction Frequent Pattern Mining Algorithm
Aggregative Clustering
Frequent Pattern Generated

**Figure 1. Formation of aggregative clustering and frequent pattern by using HDD & CFPMA**

### 3.1. Web log file

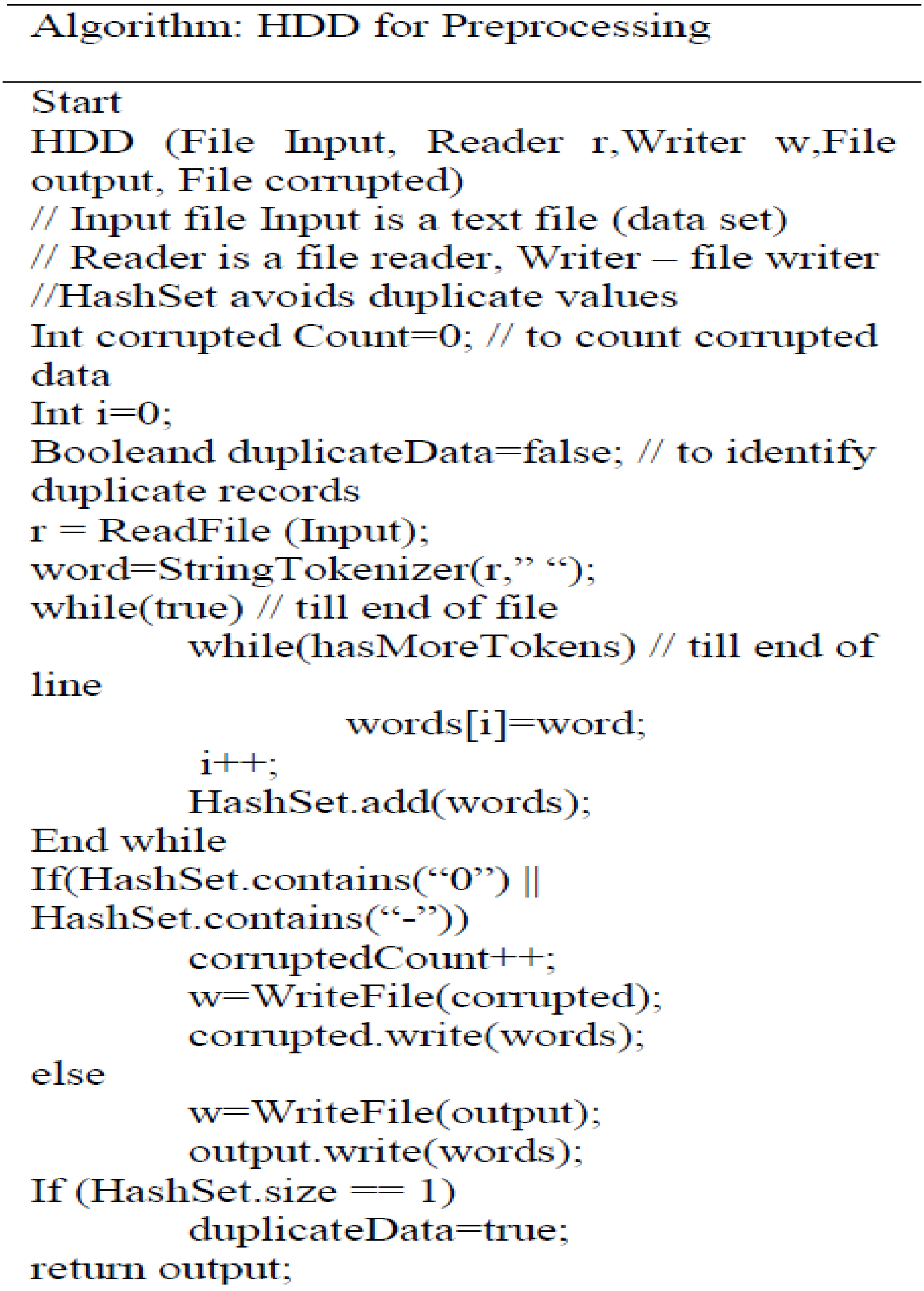
Web server log document records data about every client. At whatever point a client hits a page web server consequently gathers the log information. The log record contains an exact navigational conduct of clients, for example, name, IP address, date, time, bytes moved, get to ask for. At whatever point a client demands an asset from that specific website, relating to a HTTP demand, web server composes data in a web log document. It gives noteworthy data, for example, which pages were mentioned in a site, number of bytes sent to the client from the server and kind of mistake happens. The log record is commonly utilized for troubleshooting reason with extends from 1KB to 100MB. An example web log record is given beneath.

```
192.168.0.66--[17/Aug/2012:14:21:30-0500]”GET/jobs/HTTP/1.1”20015140
“http://www.google.com/search?q=cluster+in+data+mining&hl=en&lr=&start
=10&sa=N”“Mozilla/4.0(compatible;MSIE6.0;WindowsNT
5.1;SV1;.NETCLR1.1.4322)”
```

“192.168.0.66” is an IP address that can be converted to host name. “- -“refers the name of the remote user and login to the remote user, respectively. Both are usually omitted and replaced by a dash. [17/Aug/2012:14:21:30-0500] refers date, month, year, hour, minute, second and time zone. “GET/jobs/HTTP/1.1” refers the method, URL, HTTP protocol and its version. When a browser requests a service from a web server it returns HTTP status code in response to the request. In the proposed work Hidden Damage Data (HDD) algorithm is implemented for preprocessing. It can perform data cleaning and data filtering.

### 3.2. HDD algorithm (Hidden Damage Data)

HTTP codes and Hash set procedures are utilized in HDD for preprocessing. HTTP codes are utilized to allude distinctive status of every URL demand. HTTP code blunder 4XX is chiefly gathered in the proposed work. This HTTP code assists with recognizing Bad solicitation (400), Unauthorized (401), Forbidden (403) and Not found (404). Hash set system is utilized to give consistent time, high yield execution and streamline memory utilization. During information cleaning superfluous information and uproarious information are expelled. As the undesirable information like unimportant and repetitive information are actually anticipated and expelled, the measure of information for the information extraction process is significantly diminished. In this way, dimensionality decrease is executed for improving the precision and productivity of further information handling. HDD calculation for preprocessing is demonstrated as follows:



```
Algorithm: HDD for Preprocessing

Start
HDD (File Input, Reader r,Writer w,File
output, File corrupted)
// Input file Input is a text file (data set)
// Reader is a file reader, Writer – file writer
//HashSet avoids duplicate values
Int corrupted Count=0; // to count corrupted
data
Int i=0;
Booleand duplicateData=false; // to identify
duplicate records
r = ReadFile (Input);
word=StringTokenizer(r,” “);
while(true) // till end of file
            while(hasMoreTokens) // till end of
line
                        words[i]=word;
             i++;
            HashSet.add(words);
End while
If(HashSet.contains(“0”) ||
HashSet.contains(“-”))
            corruptedCount++;
            w=WriteFile(corrupted);
            corrupted.write(words);
else
            w=WriteFile(output);
            output.write(words);
If (HashSet.size == 1)
            duplicateData=true;
return output;
```

**Figure 2. Hidden Damage Data Algorithm for Preprocessing**

### 3.3. Aggregative Clustering

After HTTP log documents have been cleaned in information preprocessing, distinguishing proof of clients is finished. It recognizes an individual client by utilizing their IP address. Two back to back passages of the client's IP address are looked at. In the event that the IP address is same, working framework and the client's program are checked. On the off chance that both are same, the two records are considered from a similar client. Invalid client distinguishing proof is evacuated before bunching is taken care of. Bunching is a procedure of total the comparative session together. It is utilized to interest shrouded designs that exist in datasets. By utilizing the similitude measurements client bunching is applied. In view of client time, time interim and client session, session ID are executed. The arrangement of pages visited by a particular client at a particular time is known as session time. Relies upon stay time on pages the contrast between two timestamps is determined for the time interim. A lot of pages visited by a similar client inside the length of one specific visit to a site are called client sessions. During a period, a client may have solitary or different sessions. Time based likeness measurements are determined for session bunch. At last aggregative bunching is created by joining client group and session bunch. All around characterized bunches with Similar and divergent groups are gotten in the consequence of total bunching. After the ID of client and session, two sorts of grouping are applied. They are User based grouping and Session based bunching. The client closeness measurements, organize based client likeness (NBU comparability) is proposed. The bunching of the clients relying upon the system ID found in the IP or the hostname of log information is performed by utilizing the NBU closeness. After the grouping of clients, session put together bunching is performed based with respect to the time interim of the clients perused and set by the organized clients. Aggregative bunching technique is applied to join the arrangement of information by utilizing set of conditions. Two limitations, client based and session based conditions are joined to give successful groups to the web information. Along these lines, in aggregative grouping system, the obtained client and session bunch are joined. The likeness and difference of both the client and session based groups are considered. The proficiency of the information is upgraded by diminishing the unpredictable examples. The time improvement is gotten by creating the aggregative consequences of both client and separate session time interims.

### 3.4. Conviction Frequent Pattern Mining Algorithm (CFPMA)

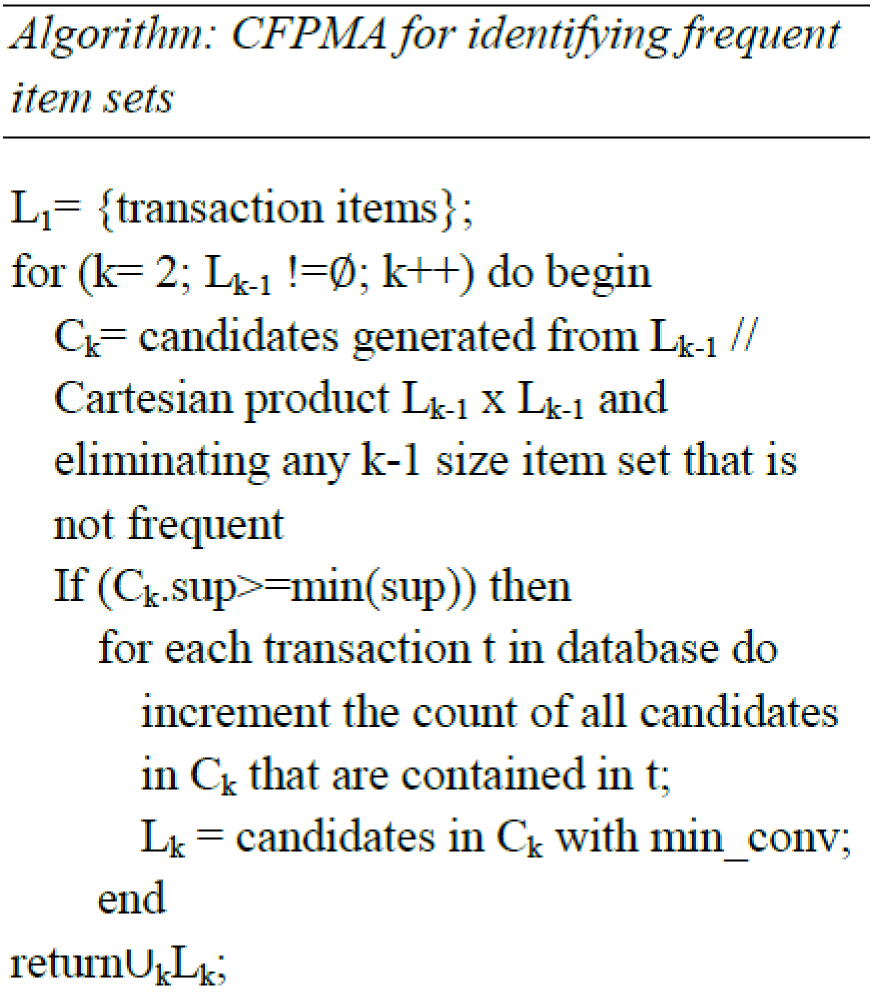
The lift and conviction values are figured for every single thing in the dataset. The lift of a standard is characterized as the proportions of the watched help to that normal if X and Y are independent. It is the proportion of the exhibition of a focusing on model (affiliation rule) at foreseeing or arranging cases as having an upgraded reaction; i.e.

$$\text{Lift}\ (X \rightarrow Y) = \frac{\text{supp}\ (X \cup Y)}{\text{supp}\ (X) * \text{supp}\ (Y)} \qquad \textbf{(1)}$$

The conviction of a rule is the ratio of expected frequency that X occurs without Y if X and Y are independent divided by the observed frequency of inaccurate predictions; i.e.

$$\text{conv}\ (X \rightarrow Y) = \frac{1 - \text{supp}\ (Y)}{1 - \text{conf}\ (X \rightarrow Y)} \qquad \textbf{(2)}$$

The minimum conviction value is set as a threshold limit to determine frequent patterns. The candidate itemsets with minimum conviction are collected. These itemsets are known as frequent patterns. The steps involved in frequent pattern generation process are shown in Figure 3:



*Algorithm: CFPMA for identifying frequent item sets*

```
$L_1$= {transaction items};
for (k= 2; $L_{k-1}$ !=∅; k++) do begin
   $C_k$= candidates generated from $L_{k-1}$ //
   Cartesian product $L_{k-1}$ x $L_{k-1}$ and
   eliminating any k-1 size item set that is
   not frequent
   If ($C_k$.sup>=min(sup)) then
      for each transaction t in database do
         increment the count of all candidates
         in $C_k$ that are contained in t;
         $L_k$ = candidates in $C_k$ with min_conv;
      end
return$\cup_k L_k$;
```

**Figure 3. Frequent Pattern Generation Process**

Here L1 denotes the set of initial dataset records. $C_k$ represents the candidate item sets that are acquired by mining least byte records by using minimum weight value. The extracted candidate item set is processed to mine frequent patterns. Before the computation of confidence, lift and conviction values, it is checked whether the support count for each item in the candidate item set ($C_k$) is greater than the minimum support count value. The candidates in $C_k$ with minimum conviction value are stored in $L_k$. So $L_k$ is the collection of frequent patterns.

## 4. Performance Analysis

This paper gives a schematic and a short description of the usage data mined from the processed log file and pattern generated. The web log data examined for evaluation is collected from the NASA Kennedy Space Center WWW server in Florida. The logs are in an ASCII file with single line per request. Two sets of web log were taken for result analysis. These two traces contain two months' worth of all HTTP requests to NASA Kennedy space center. The first log was collected from July 1, 1995 ,00:00:00, through July 31, 1995, 23:59:59, a total of 31 days. The second log was collected from 00:00:00 August 1, 1995 through 23:59:59 august 31, 1995, a total of 7 days. A prior summary is generated as soon as the web log data are loaded into the preprocessor. The two datasets consist of total number of requests of the analyzed log file, number of corrupted or failed requests, number of satisfied requests, volume of transferred bytes, etc. After preprocessing is done then by using HDD algorithm the result obtained is shown below:

**Table 1. Results of Hdd Algorithm**

| Data Analysis | Results in Count |
|---|---|
| Total no of records in Web Log | 400 |
| Date Analysis | 1 |
| Time Analysis | 244 |
| Time Zone Analysis | 1 |
| Information Analysis | 163 |
| Reply Bytes Analysis | 144 |
| Request Code Analysis | 4 |

A session of the user is created as long as the particular user is related to the website. Most of the time, default session time-out was taken as 30 minute time-out. A session would be analyzed by a user logging into a computer, performing work and then logging off. Figure 4 shows the comparison graph between user sessions for first set of log file.

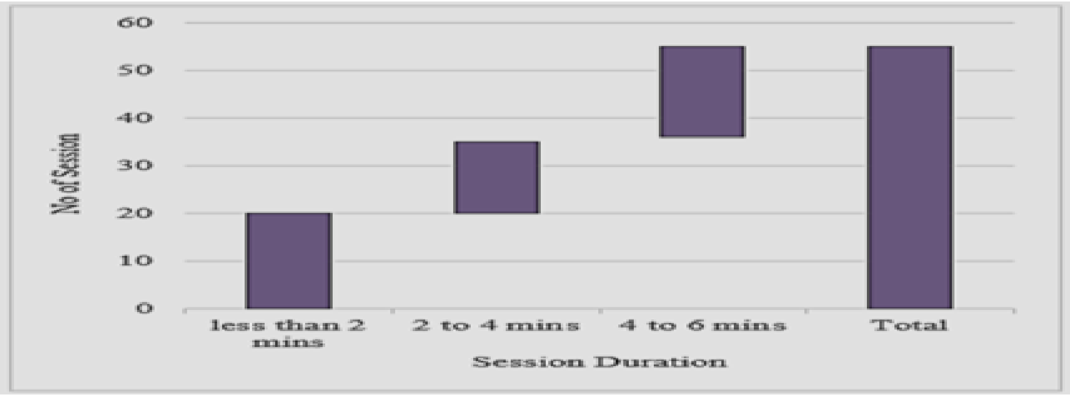


**Figure 4. Comparison Between User Sessions for First Set of Log File**

Finally, aggregate cluster is finding out by combining user and session cluster. Similar and dissimilar cluster is retrieved in the result of aggregate clustering. Figure 5 shows the aggregation of user and session cluster. For same network with same timing 15 records are obtained. For a same network with different timing 33 records are obtained.

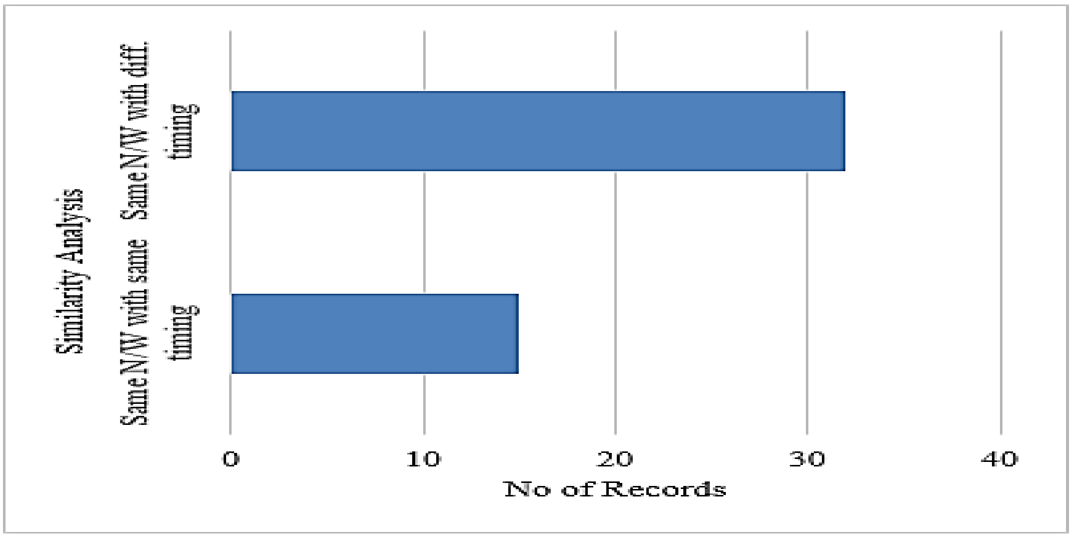


**Figure 5. Aggregation of User and Session Cluster**

The execution time consumption of the proposed approach is compared with the existing algorithms, such as AIS, SETM, and Apriori algorithm. Figure 6 & Figure 7 show the proposed CFPMA approach consumes less execution time and achieves higher accuracy when compared with the existing methods.

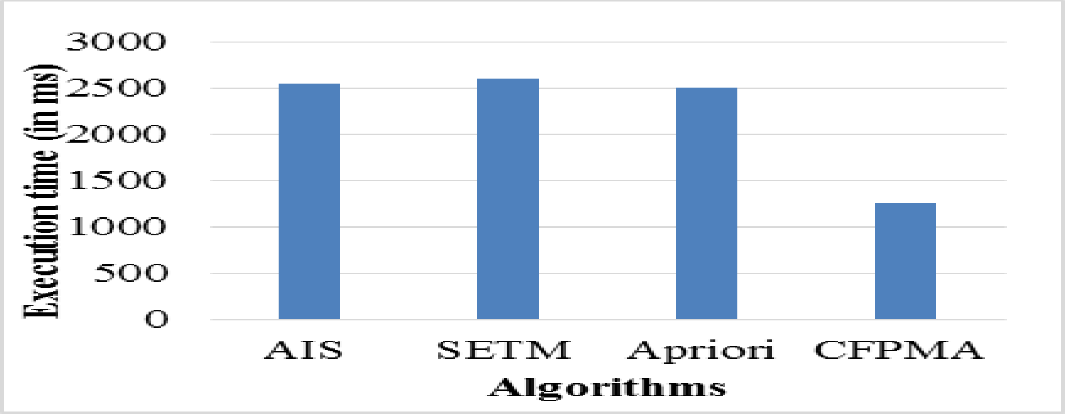


**Figure 6. Comparison of Execution Time Consumption**

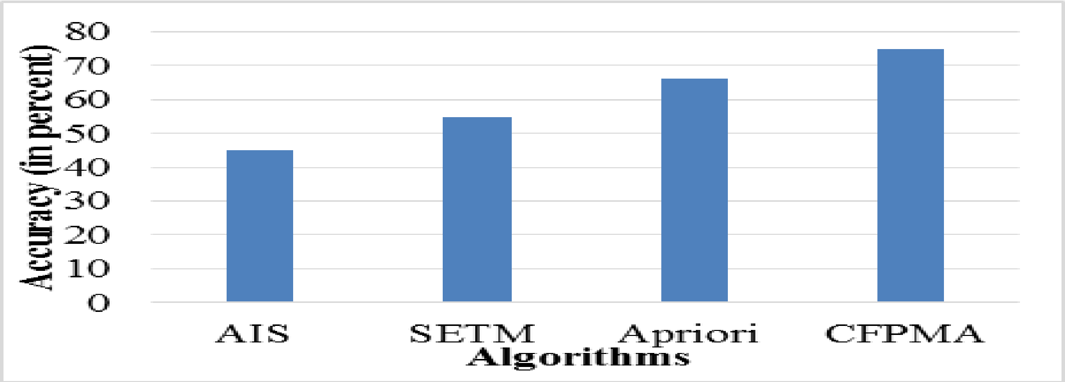


**Figure 7. Comparison of Accuracy (in percentage)**

## 5. Conclusion

In data mining, the essential task is the formulation of an appropriate target data set to which data mining and algorithms can be applied. This paper focused on web log file format, preprocessing, clustering and pattern generation techniques. Data preprocessing was implemented by using HDD algorithms to filter and organize appropriate information. Moreover, this research dealt with Conviction Frequent Pattern Mining Algorithm (CFPMA) to extract frequent patterns from the web log data. In the existing approaches, frequent pattern mining is performed based on the support count threshold limit so that, accuracy is low and the execution time is high. But in the proposed CFPMA technique, the frequent patterns are acquired based on the conviction threshold value. The experimental results gives that time taken for HDD algorithm is reduced when compared to the existing method and the proposed CFPMA technique consumes less execution time and higher accuracy than other existing approaches.

## References

[1] S. Bhattacharya, D.S. Rungta and N. Kar, “Intelligent Frequent Pattern Analysis in Web Mining”, International Journal of Digital Application & Contemporary research, Vol. 2, (2013).

[2] A. Bianco, G. Mardente,M. Mellia, M. Munaf and L. Muscariello, “Web User Session Characterization via Clustering Techniques”, Computer Networks, Special Issue on Long-Range Dependent Traffic, Vol.40, No.3, PP. 319-337, (2012).

[3] C. J. Carmona, et al., “Web Usage Mining to Improve the Design of an E-Commerce”, Website: Orolivesur.Com," Expert Systems with Applications, Vol. 39: PP. 11243-11249, (2012).

[4] M. R. Gupta and P. Gupta, "Fast Processing of Web Usage Mining with Customized Web Log Pre-processing and modified Frequent Pattern Tree", International Journal of Computer Science & Communication Networks, Vol. 1, (2011).

[5] M. R. Joel, M. Srinath, and N. Venkatesan, "An Efficient Web Personalization Approach to Discover User Interested Directories," Journal of Emerging Technologies in Web Intelligence, Vol. 6, PP. 142-148, (2014).

[6] K.-C Lin, I.-E. Liao, and Z.-S Chen, "An improved frequent pattern growth method for mining association rules," Expert Systems with Applications, Vol. 38, PP. 5154-5161, (2011).

[7] L. T. Ly, et al., "Data Transformation and Semantic Log Purging for Process Mining", in Advanced Information Systems Engineering, PP. 238-253, (2012).

[8] M. R. Mishra and M. A. Choubey, "Discovery of frequent patterns from web log data by using FP-growth algorithm for web usage mining," International Journal of Advanced Research in Computer Science and Software Engineering Vol. 2, (2012).

[9] A. K. Mishra, et al., "Web Usage Mining Using Self Organized Map," International Journal, Vol. 3. (2013).

[10] G. K. Raju and A. N. Rajimol, "A Novel Weighted Support Method for Access Pattern Mining",International Arab Journal of e-Technology, Vol. 3, PP. 201-209, (2014).

[11] S. A. Raiyani and S. Jain, "Enhance Preprocessing Technique Distinct User Identification using Web Log Usage data," International Journal of Computer Science & Communication Networks, Vol. 2, PP. 526-530, (2012).

[12] Sheetal A.Raiyani, Shailendra Jain and Ashwin G.Raiyani, "Advanced Proprocessing using Distinct User Identification in web log usage data", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 1, No 6, (2012).

[13] G. Sudhamathy and C. J. Venkateswaran, "An Efficient Hierarchical Frequent Pattern Analysis Approach for Web Usage Mining," International Journal of Computer Applications, Vol. 43, (2012).

[14] K. Sudheer Reddy, et al., "Understanding the Scope of Web Usage Mining &Amp; Applications of Web Data Usage Patterns", in Computing, Communication and Applications (ICCCA), International Conference, PP. 1-5, (2012).

[15] K. Sudheer Reddy, G. Partha Saradhi Varma and M. Kantha Reddy, "An Effective Preprocessing method for web usage mining", International Journal of Computer Theory and Engineering", Vol. 6, (2014).

[16] S. Taherizadeh and N. Moghadam, "Integrating Web Content Mining into Web Usage Mining for Finding Patterns and Predicting Users' Behaviors", International Journal of Information Science and Management (IJISM), Vol. 7, PP. 51-66, (2012).

[17] S. B. Thakare and S. Z. Gawali, "A Effective and Complete Preprocessing for Web Usage Mining", International Journal on Computer Science and Engineering, Vol. 2, PP. 848-851, (2010).

[18] B. Uma Maheswari, P. Sumathi, R. Umagandhi and A. Senthil Kumar, "A New Clustering and Preprocessing for web log mining", Journal of Computing and Communication Technologies, PP. 25-29, (2014).

[19] J. Vellingiri and S. C. Pandian, "A Survey on Web Usage Mining," Global Journal of Computer Science and Technology, Vol. 11, (2011).

[20] U. Yun and K. H. Ryu, "Approximate weighted frequent pattern mining with/without noisy environments," Knowledge-Based Systems, Vol. 24, PP. 73-82, (2011).

[21] X. Yu, et al., "Application of Closed Gap-Constrained Sequential Pattern Mining in Web Log Data", in Advances in Control and Communication. vol. 137, D. Zeng, Ed., ed: Springer Berlin Heidelberg, PP. 649-656, (2012).