COMPARATIVE STUDY OF EFFICIENT DATA CLEANING ALGORITHM, INNOVATIVE AND RAPID UNIQUE USER IDENTIFICATION ALGORITHM USING MODIFIED HASHING AND BINARY SEARCH TECHNIQUES

Ranjena Sriram[1] and Dr. S. Sheeja[2]

ABSTRACT

This study compares the proposed Unique User Identification Algorithm with other related works to prove its accuracy and efficiency. The overall study focuses on proposing a new Data cleaning and Unique User Identification processing and unique user identification algorithms for Web Usage Mining to discover and analyse the user's access pattern through mining of log files or log databases and the associated data from a particular website. Pre-processing technique is to clean the data and user identification process to identify unique users. Since number of users interacting with web sites around the world is increasing day by day, the amount of data generated and information gathered could help the organizations to improve their business according to the customers' needs and behavior. Hash function proposed in previous work lacked in accuracy and efficiency, which was further modified and fine-tuned in the next work.

The modified Hashing Function used in User Identification Algorithm is evaluated by comparing with other related works to prove its accuracy and efficiency. The modified Unique User Identification Algorithm is evaluated with various datasets from Murdoch University and Nehru College of Arts and Science from India. Various comparative analyses is also done with other related works and algorithm's to prove the efficiency of the proposed work. Significant results were produced, which is a significant authentication to this work.

I. INTRODUCTION

Internet has become an important an important source of information for all types of users across the globe. Today much emphasis is given to the data used in web applications to process meaningful information. Most of the web developers are interested to evaluate their web sites to further improve them to facilitate the users. To overcome this much preprocessing has to be dome to the data in Web Log Server to generate meaning patterns for processing. An efficient data cleaning technique and a unique user identification algorithm were proposed. A new Hashing key was framed and Binary Search techniques were used for unique user identification. Though the proposed technique performed well still some fine tuning is needed to improve the accuracy and efficiency. Modifications were done to the

[1]Research Scholar, Karpagam University, Coimbatore-641021, Tamil Nadu, India, E-mail : ranjenasriram@gmail.com,
[2]Associate Professor & Head, Dept of Computer Applications, Karpagam University, Coimbatore, Tamil Nadu, India .
E-mail : sheejaajize@gmail.com

Hashing function and the modified work is compared with various related work to prove the improved performance.

II. Web Usage Mining

Web Usage mining is a Datamining used to discover the usage patterns from the Web to understand the better needs of Web applications.

Web Usage Mining is a three phase process onsisting of

a) Preprocessing / Data Preparation: this task involves removing irrelevant data and converting data into suitable format

b) Pattern Discovery: Meaningful patterns are discovered by statistical methods, Data mining and associate rules for quick classification. Clustering and statistical methods are used for identifying unique patterns.

c) Pattern Analysis: The patterns generated are analyzed using OLAP tools, query management and smart agent based systems to remove irrelevant data, rules or patterns.

III. Sources and Types of Data

The major data source for Web Usage Mining is the server log files, which include web server access and application server logs. Apart from this information, additional data sources are essential for data preparation and pattern discovery, which include site files, Meta data, operational databases, application template and vast domain knowledge. In some cases data from client side, proxy level data collection (Internet Service Provider), and demographic data sources provided by data aggregation services are used for huge mining systems.

The data obtained from various sources can categorized into four primary groups [1].

a) **Usage Data :** The log data collected from Web server serves as the prime data for Web Usage Mining. Each hit on the Web server by the users HTTP request generates a entry on the Web log file of the web server. Each log entry consists of fields identifying the time of request, the clients IP address, user agents such as the browser information, operating system used by the client, cookies handled by user's browser and so on.

b) **Content Data :** Collection of objects and relations conveyed to the users. Most of the data are in format of text, images and structures generated from HTML and XML pages. Multimedia files dynamically generated page segments from scripts and record collection from databases, mete data, document attributes, and HTML variables. Domain oncology such as content and relationship via oncology language such as RDF or a database schema over the data contained in the operational database.

c) **User Data :**

Data collected from operational databases which include demographic information about registered users, user ratings on various objects such as product or movies, past purchases or visit histories of users as well as other explicit or implicit representations of user's interest.

IV. Framework of Proposed Study

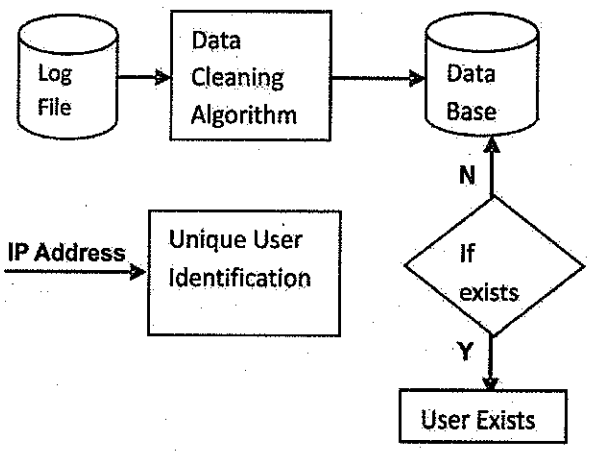

Figure 1. Framework of proposed study

V. Previous Work

An efficient Data cleaning algorithm and Unique User identification algorithm using open hashing and binary search techniques were proposed in previous work. Though the proposed Unique User Identification algorithm performed well for different web log files from different web servers, it lacked in accuracy to some extent. Considering these facts the hash function proposed in previous work is modified in this work by inclusion of some additional factors and there by proved the improved performance of the algorithm by evaluating it with different web log files from web servers.

5.1 Data Cleaning

The principle of Data Cleaning is to remove or reduce extraneous data. The following data is removed.

The principle of Data Cleaning is to remove or reduce extraneous data. The following data is removed.

- Records containing video, graphics and file extensions of GIF, JPEG and CSS.
- The log records with status codes over 299 or fewer than 200.
- Records having value of POST or HEAD.
- User agents like Crawler, Spider or Robot.

5.2 Proposed Algorithm for Data Cleaning

The generalized approach followed by earlier strategies can't be applied in real time scenarios, which are handling huge volume of data and where time is an important criterion. Considering these prevailing conditions this study introduces an innovative algorithm which follows Generalized Pattern Sequence methodology to check for irrelevant data for Web Usage Mining. Since Generalized Pattern Sequence is a mining algorithm, only the salient features are extracted for pre-processing technique.

Generally from the Web Log record the major issues considered as irrelevant data are discussed above

in the generalized algorithm. According to the modified pre-processing algorithm, different groups are created for different conditions like groups for image file extensions, methods and user agents. The input log record is fragmented and the fragments are simultaneously compared with the groups, if either one matches then the record is invalid or considered as irrelevant record and can be eliminated. The algorithm is explained below in detail [9]

5.3 Data Cleaning Algorithm using Generalized Pattern Sequence methodology proposed in previous work.

Three sequences taken for the algorithm

- File extensions like (css, jpeg, jpg, js, gif)
- Methods (GET, POST)
- Site Status (301,404,500)
- User Agents.

Input: Web server Log File

Step 1: Let F be the different Groups

k =2

Step 2: Read Log Record from Web Server Log File

Step 3: Fragment Log Record into different elements fr.

Step 4: Do while (F_{k-1} != Group Count)

Step 5: Let (a) denote individual fragments in Group F_k

For all input fragments from Log Record r in Log file (or) Database D

Step 6: If (a) matches (fr) then

Step 7: Move the record to the corresponding Group and eliminate the record from Log Database

Else move to next group

k = k +1

else

Consider the fragment as outlier.

End if

Step 8: Repeat until eof

End do

Execution of the algorithm

- Input Log Record from log File
- Generate different Groups
- Read Log Record from Log File and repeat until end of file.
- Fragment the Log Record into individual elements
- Compare each element in the groups with the input element from Log File

- If matches move the element to the individual group else move to next group.
- Eliminate the record from Log File
- Repeat the process until all groups are visited [6].

5.4. Advantages.

- Searching time minimizes since the given element from the log record is parallel checked in all groups.
- Efficient and quick when comparing with other techniques.

VI. User Identification

User's identification is, to categorize who access web site and which pages are accessed. Different users may have same IP address in the log. A referrer-based method is proposed to solve these problems in this study.

The rules adopted to distinguish user sessions can be described as follows :

- Each IP address represents one user;
- For more logs, if the IP address is the same, but the agent log shows a change in browser software or operating system, an IP address represents a different user
- Using the access log in conjunction with the referrer logs and site topology to construct browsing paths for each user. If a page is requested that is not directly reachable by a hyperlink from any of the pages visited by the user, there is another user with the same IP address.

This work comes out with an innovative Unique User Identification algorithm using Hashing techniques to locate the user in quick manner, though it is efficient it has its own drawbacks which is modified and proposed as modified algorithm which uses grouping of similar zone Ip's followed by Hashing and Binary Search techniques to locate the user more faster when comparing with this UUI algorithm.

6.1 Proposed Unique User Identification Algorithm Using Hashing Technique

Unique user identification is important process next to data cleaning. Unique users are identified based on the rules suggested in User Identification section. Though many efficient algorithms are there, many fail in accuracy and efficiency (time taken to identify users) when the size of the Log Database increases. Today's modern web servers are capable of handling terabytes of data conventional algorithms are obsolete in handling these scenarios. Considering the above facts, this study proposes an efficient Unique User Identification algorithm that uses modern Hashing techniques to identify unique user quickly inspire the huge size of the database. A new hashing key is prosed and successfully implemented in the algorithm to locate the user [7].

6.2. Hashing Techniques

For a huge database structure, it can be almost next to impossible to search all the index values

through all its level and then reach the destination data block to retrieve the desired data. Hashing is

an effective technique to calculate the direct location of a data record on the disk without using index structure.

Hashing uses hash functions with search keys as parameters to generate the address of a data record [5].

Hash Organization.

Generally a hash stores data in the form of a bucket, a bucket is a representative of a storage block which stores one complete disk block, which in turn stores record groups. Searching in Hash table is done by a Hash Function which maps all set of sear keys (K) to the address where the actual records are placed. It is a function from search key to bucket addresses [5].

Dynamic Hashing

The problem with static hashing is that it does not expand or shrink dynamically as the size of the database grows or shrinks. Dynamic hashing provides a mechanism in which data buckets are added and removed dynamically and on-demand. Dynamic hashing is also known as extended hashing [6].

Hash function, in dynamic hashing, is made to produce a large number of values and only a few are used initially.

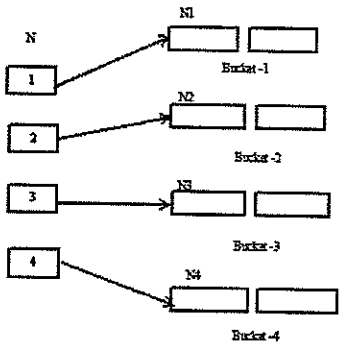

Figure 2. Dynamic Hashing Technique

Generally any Unique User Identification algorithm analyses more factors like users IP addresses, web site topology, browser edition and operating system. The proposed algorithm not only uses IP addresses but also identifies user's session. The proposed algorithm not only uses User IP address, but also based on path chosen by any user, access time with the referred page etc. [4].

When huge databases are taken for considering the time taken to locate the records is much, hence appropriate methodology is incorporated to make the process faster. Taking these prevailing conditions, the study proposes a new Hashing formulation, to minimize the searching time for large datasets. Previous study proposed a Hashing function, which is quick enough to search the unique user's IP address, but when the size of the bucket increases certain pre-processing is done to fasten the searching time of unique user.

On considering the prevailing issues, this work substitutes Binary search techniques to minimize the searching time.

A few modifications are done the previously proposed Hashing function to make it generalized and quick in searching patterns.

Hash Function proposed in previous work.

$Nmod_2 * K + d + "E$ (1.1)

N - refers to the record number

K - Virtual Address of the hash bucket

D - Displacement distance

"E- is the new error factor.

ΔE = Original memory address – Address obtained from Hashing function.

The generalized function by inclusion of error factor is H(K) + "E. (1.2)

The error function removes the discrepancy obtained from the hash function and helps to obtain the exact address required to fetch the user record.

Advantages of the new Hashing function.

1) Accurately locates and retrieves the data from the memory location.
2) Increases the accuracy and efficiency.
3) Improves the overall efficiency of Unique User Identification Algorithm

a. 6.3 .Proposed Unique User Identification Algorithm

Web Log server contains accumulated log information, which makes the searching more complicated. In order to minimize the searching time this work includes two main strategies.

An algorithm is designed and developed to group IP addresses of similar zones in individual Hash buckets from the Web Log Server file, which minimizes the searching time to a great extent.

Binary Search techniques are used along with some string manipulations in the previously proposed UUI algorithm to reduce the searching time.

Different sets of IP ranges are allocated to particular networks, geographic areas, companies etc. The table below shows several examples of IP ranges and their implementations.

6.3. First Strategy

Table 1. IP addresses of different zones

IP Range	Description	Example
192.168.——	Private Networks	192.168.1.23
172.16.——-172.31.——		
-		
10.——		
41.——	AfriNIC allocation of IP	102.43.1.65
102.——	addresses in Africa	
105.—		
81.—	European allocations of IP	81.202.37.89
217.—	addresses	
62.—		
200.	Latin American and	200.180.50.25
	Caribbean	
9.	IBM	9.1.2.3
17.	Apple	17.13.5.13

Since different zones start with different IP address the IP address of the given user is searched must be differentiated and grouped to their specific zones in separate Hash buckets assigned for different IP zones from other zones of IP addresses. The algorithm explained below shows how specific IP address from a particular zone is extracted and stored in separate Hash bucket in the form of array. This tactic reduces the searching time to much extent which is shown in the result and discussions section.

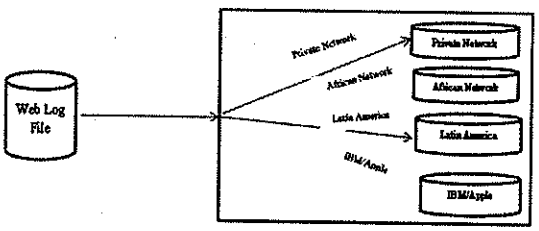

Figure 3. Grouping of IP addresses from Web Log Server in separate Hash buckets using prosed SetIp algorithm

SetIp Algorithm to group IP zone wise from Web Log Server File

```
Unique User Identification (UUI)
Definition: given a clean and filtered web log file and record set web log file
Records R= {r1,r2,r3 ..... rn}
where n>0
Step1:  input Log database RUser of N records
Step2:  Distinct User identification base
Step3:  RUser=P<url, ip_addr, agent, method, operating system,status,session id,time_stamp>
Step4:  RUser=<r1,r2,r3....rn> where n!=0,i=0
Step5:  while (i<n)
Step6:  GetIp(i) = substr(R(i),0,3  // Obtain first three digits of IP Address
Step 6: while (Log database<>eof)
Step 7: read Log database RUser
Step 8: switch (GetIp(i))
   Begin
      Case 192 :        BinarySearch(PN, R(i) *( Rand1 * K(i) + d)).
                        If found extract user information Else //Existing User
                        Store the IP address to Private Networks Array PN //New User
                        Assign the PN Array to Hash Storage.
                        end if
                        Break.
      Case 41, 102, 105: BinarySearch(AN, R(i) *( Rand1 * K(i) + d)).
                        If found extract user information Else
                        Store the IP address to African address Array AN
                        end if
                        Assign the AN Array to Hash Storage.
                        Break.
      Case 81, 217, 62: BinarySearch(EN, R(i) *( Rand1 * K(i) + d)).
                        If found extract user information Else
                        Store the IP address to European IP addresses Array EN
                        Assign the EN Array to Hash Storage.
                        end if
                        Break
      Case 200:         BinarySearch(LA, R(i) *( Rand1 * K(i) + d)).
                        If found extract user information Else
                        Store the IP address to Latin American IP addresses Array LA
                        Assign the LA Array to Hash Storage.
                        Break.
      Case 17:  BinarySearch(AP, R(i) *( Rand1 * K(i) + d)).
                If found extract user information Else
                Store the IP address to Apple IP addresses Array AP
                Assign the AP Array to Hash Storage
                end if

Step 9:  End Switch
Step10:  end loop (Log database)
Step11:  i=i+1;
Step12:  end loop (Web log file)
Step13:  end
```

Figure 4 Algorithm GETIP to extract IP address of Unique User.

6.4. Execution of SetIp Algorithm

- Read records one by one until end of file
- Get first three digits from the each record.
- Match with the case statement.
- If the record matches either of the case statement, store the IP in either of the arrays assigned to each zone.

6.5. Second Strategy

Now this arrangement facilitates the modified UUI algorithm to search whether the IP of the user exists or not quickly. In order to achieve this task some modifications are done to the previously proposed UUI algorithm.

· Binary Search techniques are used along with some string manipulations in the previously proposed UUI algorithm to reduce the searching time.

```
Unique User Identification (UUI)
Definition: given a clean and filtered web log file and record set web log file
Records R= {r1,r2,r3.....rn}
where n>0
Step1:  input Log database RUser of N records
Step2:  Distinct User identification base
Step3:  RUser=P<url, ip_addr, agent, method, operating system,status,session id,time_stamp>
Step4:  RUser=<r1,r2,r3...rn> where ri!=0,i=0
Step5:  while (i<n)
Step6:  GetIp(i) = substr(R(i),0,3  // Obtain first three digits of IP Address
Step6:  while (Log database<>eof)
Step7:  read Log database RUser
Step8:  . switch (GetIp(i))
    Begin
        Case 192 :      BinarySearch(PN, R(i) *( Rmod; * K(i) + d)).
                        If found extract user information Else //Existing User
                        Store the IP address to Private Networks Array PN //New User
                        Assign the PN Array to Hash Storage.
                        end if
                        Break.
        Case 41, 102, 105: BinarySearch(AN, R(i) *( Rmod; * K(i) + d)).
                        If found extract user information Else
                        Store the IP address to African address Array AN
                        end if
                        Assign the AN Array to Hash Storage.
                        Break.
        Case 81, 217, 62: BinarySearch(EN, R(i) *( Rmod; * K(i) + d)).
                        If found extract user information Else
                        Store the IP address to European IP addresses Array EN
                        Assign the EN Array to Hash Storage.
                        end if
                        Break
        Case 200:       BinarySearch(LA, R(i) *( Rmod; * K(i) + d)).
                        If found extract user information Else
                        Store the IP address to Latin American IP addresses Array LA
                        Assign the LA Array to Hash Storage.
                        Break.
        Case 17:    BinarySearch(AP, R(i) *( Rmod; * K(i) + d)).
                    If found extract user information Else
                    Store the IP address to Apple IP addresses Array AP
                    Assign the AP Array to Hash Storage
                    end if

Step9:  End Switch
Step10: end loop (Log database)
Step11: i=i+1;
Step12: end loop (Web log file)
Step13: end
```

Figure 5. Unique User Identification (UUI) Algorithm

6.6. Execution of the Algorithm

- Get the input user IP
- Extract the first three digits from it
- Check to which zone it belongs using the switch statement.
- If it matches a particular zone, then search the given user IP in that particular zone Hash bucket using Binary Search and Hash function.
- If found extract the user information, if not assign the IP address to that [articular zone and treat it as new user.

6.7. Advantages of the modified UUI Algorithm

- Since the IP addresses are grouped zone wise, easy to search and locate the users IP addresses and their relevant information.
- Binary Search techniques combined with Hash function makes the searching faster minimizing the time.
- This proposed algorithm proves and shows better results over other UUI algorithms, which are elaborated in the results and discussions section.

VII. Results and Discussions

To evaluate the performance of the proposed work various related studies done by other scholars on data cleaning and Unique User Identification algorithms.

Windows 2000 professional, SQL Server 2000 and MATLAB (7.9.0.529). MATLAB tool is used to

develop applications to evaluate the performance of the proposed algorithms. The results obtained prove the better and improved performance of the proposed algorithms over other related works.

The proposed modified UUI Algorithm is compared with the works done by Shetal.A.Raiyani's study published in (International Journal of Computer Science & Communication Networks, Vol 2 August 2015) [9] and K.R.Suneetha and Dr.Krishnamoorthi study published in (International Journal of Computer Science and Network Security, Vol 9 No 4 April 2009 [11]). The results obtained are displayed in the table below.

Table 2. Performance of Modified UUI Algorithm with other related works.

Performance Analysis	Database Source	Record Size	Entries in Raw Web Log	Entries After Data Cleaning	No of Users	No of Unique Users	Execution Time(s) Previous Work	Execution Time(s) Modified Work.
Unique User Identification Algorithm Proposed by Sheta.A.Raiyani	Web Server Log R.K.University	10^7	47890	12783	6542	4366	0.2567	NIL
Proposed Modified Unique User Identification Algorithm	Web Server Log MURDOC University, Dubai	10^{12}	1030002799 00	1000000027 83	567502876	436675 422	0.4247	0.3582
Unique User Identification Algorithm proposed by K.R.Suneetha and Dr.R.Krishnamurthy.	NASA Server Log	10^4	87233	33657	4000	1765	0.5432	NIL
Proposed Modified Unique User Identification Algorithm	Web Server Log Nehru Arts and Science College	10^{12}	1256440002 77	1124000277 1	577502876	446675 422	0.4211	0.3724

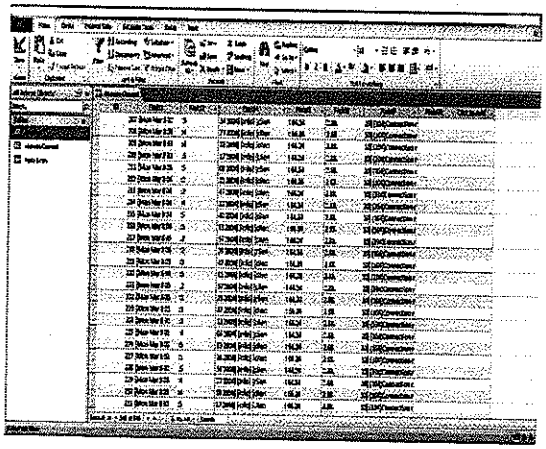

Figure 8 . Number of Web Log Server Records of Murdoch University before implementation of proposed Data Cleaning Algorithm

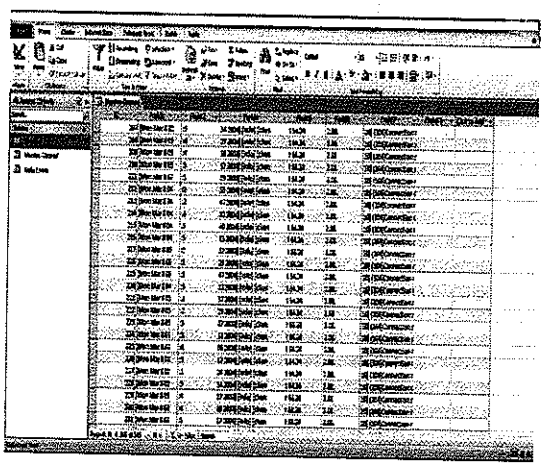

Figure 9. Number of Web Log Server Records of Murdoch University after implementation of proposed Data Cleaning Algorithm

From Table 1 it is clearly evident that the modified UUI algorithm is far better than the older algorithm and algorithms prosed by Shetal.A.Raiyani in their work. The modified algorithm shows much clarity in data cleaning and also proves in its efficiency by the consuming less execution time. It takes only 0.3582 second to identify the number of unique users for data size of 10^{12} records, whereas the referred algorithm consumes 0.2567 and 0.5432 seconds for data sizes of 10^3 and 10^4 seconds. For data size of 10^3record sizes, UUI algorithm proposed by K.R.Sangeetha dn Dr.Krishnamurthy tconsumes 0.5432 seconds to identify 1765 unique users whereas the proposed modified UUI algorithm takes 0.3724 seconds to identify 446675422 unique users , which proves that the modified UUI algorithm takes less time to execute inspire of the huge data size. Still work is in progress to fine tune the algorithm and improve its efficiency to an appreciable extent

From the above Figures 8 and 9, it is evident that the proposed Data Cleaning algorithm has performs well. Sample of 642 records were taken from MURDOC University Web Log Server and the data were cleaned using the proposed Data Cleaning algorithm, interestingly 342 irrelevant records were eliminated at a time factor of 1.25 (s). This result is a valid proof for the performance of the proposed Data Cleaning Algorithm.

VIII. Conclusion

This paper has come out with a modified version of UUI algorithm with modified hash function to identify unique users. The older version of UUI algorithm proposed lacked accuracy to some extent. To resolve this problem hash function proposed in previous works has been modified by inclusion new error factor, which further improved the accuracy and efficiency of the algorithm to a great extent. Various evaluation and comparisons were made to prove the improved performance of the algorithm. The results obtained prove the verdict.

This paper has also comes out with a unique strategy to group IP addresses according to their zone specification. From the grouped IP address, this work uses Binary Search technique to locate the IP of unique user. Inclusion of this strategy in the previously proposed DUI algorithm drastically minimizes the time to search and locate users IP addresses. The algorithm is evaluated with various related studies done by eminent scholars with different universities web log server's data to identify the efficiency of cleaning process, to check number of users visited the pages, time taken to identify unique users etc. The algorithm proves and shows much improvement over the previous and other related works. Further improvements are needed to combine the whole process of Web Usage Mining. A complete methodology that covers pattern discovery and pattern analysis will be more useful in user identification process. This work helped the site developers to analyze their sites and also helped then to identify the user types and range of users. It also guided them in further redesigning their sites according to the users requirements.

IX. Reference

1. BamshadMobasher, Robert Cooley,Jaidep Srivastave. Automatic Personalization based on Web Usage Mining. Communications of ACM Volume 43, Issue 6 Aug 2000,Page(s):142-151.

2. K.SudeerRedy. An effective data pre-processing method for Web Usage Mining. IEEE, ISBN978-1-4673-5786-9, Page(s):7-10.

3. Jaideep Srivastava * t, Robert Cooley:I: , Mukund Deshpande, Pang-Ning Tan .Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data University of Minnesota 200 Union St SE Minneapolis, MN 55455.

4. Shuyan Bai ,Yantai , Qingtian Han , Qiming Liu , Xiaoyan Gao.Research of an Algorithm Based on Web Usage Mining. IEEE, ISBN: 978-1-4244-3893-8, Page(s): 1-4.

5. Review and Analysis of Hashing Techniques", International Journal of Advanced Research in Computer Science and Software Engineering", Volume 4, Issue 5, May 2014, Page(s): 296-297.

6. Ranjena Sriram. Dr.R.Mallika.Innovative Pre-Processing Technique and Efficient User Identification Algorithm for Web Usage Mining.International Journal of Advanced Research in Computer Science and Software Engineering.Volume 6, Issue 2, Feb-2016.Page(s):85-90.

7. Sangeeta Raheja .Comparative study of Hashing Algorithm Using Cryptographic and Steganography Using Audio Files. International Journal of Advanced Research in Computer Science and Software Engineering", Volume 4, Issue 5, May 2014, Page(s):292-294.

8. Sheetal A. Raiyan. Advanced Pre-processing using Distinct User Identification in web log usage data. International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 6, August 2012 Copyright to IJARCCE www.ijarcce.com 418, Page(s) : 418-420.

9. Satpal Singh .An Exclusive Survey on Web Usage Mining For User Identification. International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 11, November 2014 Page(s):6582-6586.

10. Shetal.A.Raiyani .Preprocessing and Analysis of Web Server Logs. International Journal of Computer Science & Communication Networks, Vol 2 August 2015, Page(s): 46-55.

11. K.R.Suneetha and Dr.R.Krishnamoorthi. Identifying User Behavior by Analyzing Web Server Access Log File. International Journal of Computer Science and Network Security, Vol 9 No 4 April 2009, Page(s) :327-332.

Author's Biography

Ranjena Sriram is pursuing her Ph.D degree in Computer Science at Karpagam University, Coimbatore. Currently working as a Creative Design and Innovation teacher for Ministry of Education, Dubai, United Arab Emirates. During the tenure she had represented various reputed Universities and Colleges in U.A.E in different portfolios to develop and reform Academic activities. She has presented papers and attended many International conferences to her credit.

Dr. S. Sheeja, is currently working as Associate Professor and Head of the Department of Computer Applications at Karpagam University, Coimbatore. She had completed Ph.D in Computer Science at Bharathiar University in 2015. She has more than 13 years of teaching experience to her credit. Her primary research interests are related to Computer Networks, Mobile Adhoc Networks, Mobile Computing, Image Processing and Data mining. She has published papers in various International and national Journals.