International Journal of Engineering & Technology, 7 (1.9) (2018) 54-63

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Efficient data cleaning algorithm using decision tree classification model approach and modified new unique user identification algorithm using hashing techniques with a new error factor

Ranjena Sriram [1] *, S. Sheeja [2], I. Henry Alexander [3]

[1] Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore
[2] Associate Professor & Head, Dept of Computer Applications, Karpagam University, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India
[3] Research Consultant, Udumalpet, Tirupur district, Tamil Nadu, India
**Corresponding author E-mail: ranjenasriram@gmail.com*

Abstract

The study focuses on preprocessing techniques of web mining. Considering this scope, the study has proposed and implemented an efficient data cleaning and unique user identification algorithms. Previously proposed data cleaning algorithm is a generalized approach and lacked transparency. An appropriate model has to be used to implement the new data cleaning algorithm. Over analysis of various related studies and suggestions made by eminent experts, the study finalized decision tree classification model, and appropriate model to implement the new data cleaning algorithm. Simplicity, ease in framing rules and ability to fragment complex decisions to solve a problem motivated to choose decision tree classification model to implement new data cleaning algorithm. Apart from this the study has also modified the previously proposed hash function, used to locate existing web users in web log server. A new error factor is introduced to remove memory address discrepancy. The modified hashing function along with binary search techniques is used to design the new unique user identification algorithm. Various experiments analysis is done using web log servers of eminent universities and colleges from United Arab Emirates and India. Results obtained prove the improved and better performances of the new rule based data cleaning and modified unique user identification algorithms.

Keywords: *Use about five key words or phrases in alphabetical order, Separated by Semicolon.*

1. Introduction

Internet has become an important source of information for all the types of users across the globe. Today more importance and emphasis is given to data used for any study irrespective of the domains. Relevant data is needed for any study to produce accurate results and meaningful patterns in case of mining process. Hence this study focuses on two important preprocessing techniques of web usage mining process. Data cleaning and unique user identification are the two major preprocessing tasks considered for this study. To produce minimum error free data, guiding and generating summary to the organizations to self-evaluate their web sites are the main scope of this study. Web mining, web usage mining and data mining concepts are integrated to make this study feasible.

Web Usage mining is a Datamining used to discover the usage patterns from the Web to understand the better needs of Web applications. It extracts information from surfer's session for further processing. Web content and web structure make use of the primary data while Web Usage mining use the secondary data from the Web Log files stored in the Web Servers. [1].

Web Usage Mining is a three phase process consisting of

a) Preprocessing/Data Preparation – Meaningful and proper dataset are needed to derive meaningful patterns. Getting relevant data from the various sources in the present world is a major task. Improper and irrelevant dataset will result to inaccurate and poor results. Studies are still in progress for overcome this problem. Hence much emphasis is given to this phase to get accurate data. Data preprocessing is the most difficult task in the mining process. Efficient, robust and versatile algorithms from other domains are used in this process to derive meaningful patterns.
b) Pattern Discovery: Patterns are generated using the Statistical Data mining, Associate rules and Sequential methods. These strategies make classification fast and efficient. Statistical methods, Data mining methods, Associate rule, Sequential methods and cluster techniques are used to identify unique patterns. Unique patterns derived from implementing these strategies helps to derive accurate and efficient results after mining.
c) Pattern Analysis: The patterns generated are analyzed using the OLAP tools, query management and smart agent based systems to remove irrelevant data, rules or patterns. This process improves the accuracy of the data, which further results in accurate mining [1].

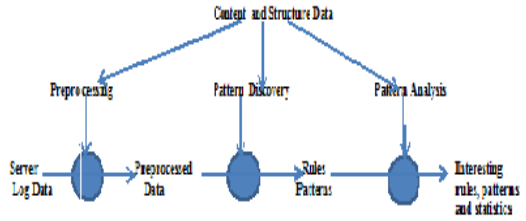

Fig. 1: Process of Web Usage Mining.

2. Sources and types of data

The major data source for the 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 the 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 provide by the data aggregation services are used for huge mining systems [1] [2].

The data from the various sources can be categorized into four primary groups. a) Usage data b) Content data and c) User data.

a) Usage data is the main source of data for web mining. Log records are stored in web log servers after each hit on the server by the users, serves as the central data, which is further mined to generate meaningful patterns. The log record contains useful information like users IP address, time of request, information about various agents like users browser information, user's operating systems and cache details. This information guides the Web developers to further improve their sites.

b) Content Data: Collection of objects and relations are conveyed to the users. Most of the data are in the format of text, images and structures generated from HTML and XML pages. Multimedia files dynamically generate page segments from scripts and record collection from the 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: Demographic information of registered users collected from operational databases, user ratings on different category of products such as movies, past purchases and details if visit histories of web users and other features or representations of user's interest cover the user data. These information help to generate summaries to guide and self-evaluate several organizations and web developers.

3. Previous work

An efficient Data Cleaning Algorithm along with an innovative Unique User Identification Algorithm was proposed in previous study. Several factors and web attributes were considered to design and implement the new data cleaning algorithm.

3.1. Data cleaning

The principle of Data Cleaning is removed or reduces extraneous data. The following data are removed.

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

3.2. Proposed algorithm for data cleaning

Several data cleaning algorithms proposed earlier by eminent scholars were a general explanation which was quite difficult to implement. They also failed to handle huge datasets. A suitable model has to be used to implement this algorithm. Considering these facts an efficient data cleaning algorithm was proposed in previous study.

Individual groups for file extensions, server request, response methods, web site status and user agents are the main attributes used in the algorithm. Each log record is fragmented and the above mentioned information is extracted from them. 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. [13] [14].

3.3. 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 the 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.

3.4. 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.

3.5. Disadvantages

- Generalized logic

- No much clarity in logic
- Uses simple if then else statements to match criteria.
- No specific model implemented to the model.

3.6. Analysis of decision tree classification model

A decision tree is a flowchart like tree structure, where each non leaf node represents a test on the selected attribute and the branches represent the outcomes of the test. The leaf node represents the classified labels [7] [8].

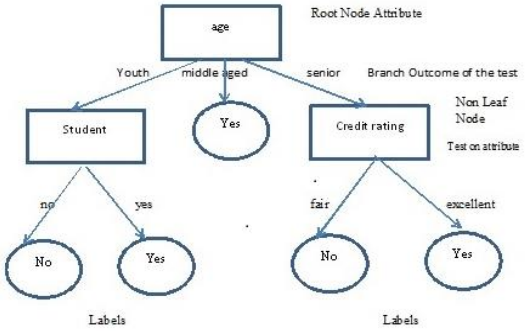

Fig. 2: Decision Tree Classification Model.

3.6.1. Advantages of decision tree classification model

- Decision tree model doesn't require any domain knowledge or parameter setting.
- Decision tree can handle high dimensional data.
- Easy to assimilate by users.
- The learning and classification steps are simple and fast.
- Fast in accuracy.

3.6.2. Decision tree classification model and proposed data cleaning algorithm

File extensions, status codes, server methods and user agents ate the attribute list based on which the records are eliminated.
A sample attribute splitting and identifying good and bad tuples are displayed below in Figure.3.

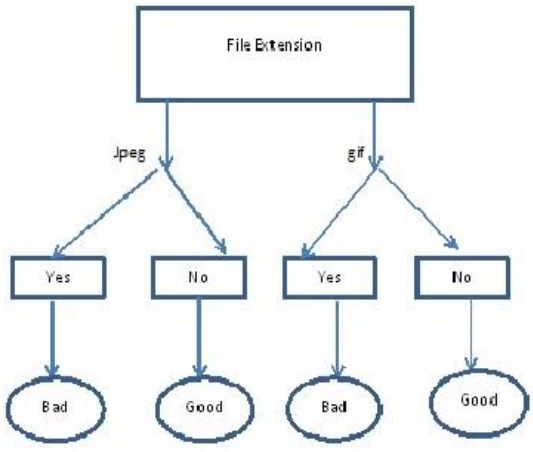

Fig. 3: Data Cleaning Using Decision Tree Methodology.

Figure 3 explains how an irreverent record is removed from the web log file. Based on the criteria's for removing irreverent record, attribute (file extension) is considered. File extension attribute is placed on the root node and the branches of the node have outcomes of the extensions which are all image and server script based file extensions. The non-leaf node contains the test on the attribute to check whether the file extension ends with an image or server scripting file extensions. If the value is (Yes) then the leaf node will contain the label (GOOD) else it will contain the label (NO).

First methodology of attribute splitting that is branch created from the root node based on the known value of the attribute is used to select and appropriate attribute for the model [9] [10].
Example of a simple rule based on which the record is eliminated.

If file extension = "jpeeg" (V) File extension = "jpg" (V) File Extension = "bmp" (V) File Extension = "css" = Bad Record

Similar rules are created for all the criteria based on which the outcomes and tests are done over various attributes.

3.6.3. New data cleaning algorithm using decision tree classification model

Algorithm Generate Decision Tree (Generate a decision tree from training tuples of data D)

Input

Data partition D, which is a set of training tuples and associated class labels.

Attribute List, set of log record attributes.

Attribute selection method, a procedure to determine the splitting criteria

Method

Step 1: create a node N

Step 2: I tuples in D are of same class C then

Step 3: Return N as the leaf node which contains class label

Step 4: If Attribute list is empty then

Step 5: return N as the leaf node

Step 6: Apply Attribute selection method

Step 7: Label N with splitting attribute

Step 8: if splitting attribute is discrete valued and

Multiday split allowed then

Step 9: attribute list – attribute list – splitting value

Step 10: for each outcome j of splitting criteria

Step 11: let D_j be the set of data tuples in D satisfying outcome j.

Step 12: If D is empty then

Step 13: attach a leaf labelled with majority class in D to N_j

Step 14: else attach the node returned by Generate Decision Tree (D_j, attribute list) to Node N.

End for

Fig. 4: Data Cleaning Algorithm Using Decision Tree Classification Model.

3.6.4. Execution of modified data cleaning algorithm using decision tree classification approach

The input for the algorithm is data partition, set of associated training tuples, class labels, attribute list and an attribute selection method. First step an empty node is created. If all the tuples in (D) are same no need for partitioning the node becomes the class label, else the tuples are separated based on attribute splitting. This is done by the applying suitable attribute selection method. More than one outcome is generated from the selected attribute. For each attribute matching tuples are selected from (D). The algorithm is stopped until no outcomes are generated (or) if all the tuples in (D) are classified. The output is a generated tree model.

3.6.5. Advantages of modified data cleaning algorithm using decision tree classification model

- Fast in taking decision.
- Easy to frame rules.
- Capable of handling huge datasets.
- Increased accuracy.
- Increased efficiency by removing more irrelevant data over other data cleaning algorithms.
- More transparency in its execution.

4. Modified unique user identification algorithm

4.1. Modified hashing function

Hash Function proposed in previous study.

$$\text{Ceil } (N \text{ div}_2) * K + d \quad (1.1)$$

Where N refers the record number indirectly pointing the data an IP address or an Operating system or a browser, (K) refers to the virtual address of the bucket and d refers to the displacement distance. The multiplied factor gives the original location of the data [31] [52].

Drawbacks

- Takes more time to locate the required user.
- Some pre-processing required minimizing the searching time.
- Takes user records (IP addresses) as such without splitting zone wise hence more time in identifying users.

Considering these drawbacks the Hash function specified in equation (1.1) is made more specific by inclusion of error factor ΔE.
Now equation 1.1 is transformed to

$$N\text{div}_2 * K + d + \Delta E \quad (1.2)$$

Where ΔE is the new error factor.

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

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

4.1.1. Advantages of the new hashing function

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

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 [11].

4.2. Modified unique user identification algorithm using modified hashing technique with proposed new error factor

Unique user identification is an important process next to data cleaning. Unique users are identified based on the rules suggested in the 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 modified hashing function with new error factor and binary search techniques to identify existing web user quickly. The algorithm also generates summary of reports which guides the organizations and web developers to self-evaluate their web applications for further amendments to increase their business [12] [13] [14].

```
Unique User Identification (UUI)
Definition: given a clean and filtered web log file and record set web log file
Step1: Input UserIP address to search the User Record
Step 2: GetIp = substr((UserIP, 0,3) // Obtain first three digits of IP Address
Step3 switch (GetIp)
Begin
    Case 192:    Binary Search (Ceil (UserIP mod2) * K + d + ?E, PN)
                 If Found extract user information Else (Existing User)
                 Store the IP address in Private Networks Array PN //New User
                 Assign the PN Array to Hash Storage
                 End if
                 Break
Case 41, 102, 105: Binary Search (Ceil (UserIP mod2) * K + d + ?E, AN)
                 If Found extract user information Else (Existing User)
                 Store the IP address in African Address Array AN //New User
                 Assign the AN Array to Hash Storage
                 End if
                 Break
Case 81, 271, 62: Binary Search (Ceil (UserIP mod2) * K + d + ?E, EN)
                 If Found extract user information Else (Existing User)
                 Store the IP address in European IP Addresses Array EN //New User
                 Assign the EN Array to Hash Storage
                 End if
                 Break
    Case 200:  Binary Search (Ceil (UserIP mod2) * K + d + ?E, LA)
                 If Found extract user information Else (Existing User)
                 Store the IP address in Latin American Networks Array LA //New User
                 Assign the LA Array to Hash Storage
                 End if
Step 4: End Switch
Step 5: end loop (Log database)
Step 6: i=i+1;
Step 7: end loop (Web log file)
```

4.3. Execution of the algorithm

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

4.4. 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 hashing techniques minimizes the searching time of existing web user.
- This proposed algorithm proves and shows better results over other UUI algorithms, which are elaborated in the results and discussions section.

5. Results and discussions

To validate the effectiveness and efficiency of the algorithms proposed, an experiment with the web server logs of Murdoch University and Emirates College of Management and Information Technology, Dubai and Nehru Arts and Science College, India, is made. This work has proposed a effective Hashing technique,

which minimizes the storage and eradicates collision problem. These changes have drastically improved the searching time of the user record and thereby improve its performance. Results obtained from the previous work are compared with the updated work. This work proves with better results to validate the work done. The initial data source of our experiment is from JAN 1, 2014 to Aug 3, 2015, with data size of 10^{12} records. The experiments are performed on a 2.8GHz Intel Celeron I, CPU, 2.00 GB of main memory, 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 table listed below illustrates the overall performance of UUI algorithm.

5.1. Various comparison evaluations

5.1.1. Comparison of new rule based data cleaning algorithm with previously proposed data cleaning algorithm

Table 1: Comparison of Rule Based Data Cleaning Algorithm with Previous Studies

Related Study	Data Source	Number of Records	Number of Fields	Missing Value (%)	Duplicate Records
Data Cleaning Algorithm version 1.0	Nehru Arts and Science College	10000	20	5.97	95
	ECMIT		18	5.94	93
Data Cleaning Algorithm version 2.0	Nehru Arts and Science College		20	3.23	110
	ECMIT		18	4.21	115
Proposed Rule Based Data Cleaning Algorithm version 3.0	Nehru Arts and Science College		20	2.47	150
	ECMIT		18	3.01	143

Table 1 displays the performance metrics of the modified unique user identification algorithm using decision tree classification model approach over the previous versions of the algorithm. From the results obtained it is evident that the modified algorithm performs better over the previous algorithms. The new algorithm has removed 150 records and 143 records over a sample of ten thousand records for Nehru Arts and Science College and ECMIT respectively, which is far better over the previous versions.

5.1.2. Comparison of rule based data cleaning algorithms with other referred rule based decision tree algorithms

Confusion matrix techniques were used to measure accuracy of the proposed new Rule Based Cleaning Algorithm.

		Predicted Class	
		Class 1	**Class 2**
Actual Class	**Class 1**	True Positive	True Negative
	Class 2	False Positive	False Negative

Fig. 5: Confusion Matrix Table.

True Positive (TP)-correctly predicted of positive classes
True negative (TN)-correctly predicted of negative classes
False Positive (FP)-negatives in correctly predicted negative values
False Negative (FN) -negatives in wrongly predicted negative values.
True Positive Rate (TPR)-positives in correctly classified positive values.
False Positive Rate (FPR)-negatives in correctly classified negative values.

- Accuracy -It shows the total number of instance prediction which are correctly predicted.

A= (TP + TN) / N

- Receive Operating Characteristic (ROC) – It is a graphical approach for displaying the trade of between true positive rate (TPR) and false positive rate (FPR) of a classifier. TPR is plotted along (y) axis and FPR along the (x) axis.
- Precision (P) – It determines exactness. It is the ratio of the predicted positive cases that were corrected to the total number of predicted positive cases.

P = TP / (TP + FP)

- Recall (R) – It determines the completeness. It is the proportion of positive cases that were correctly recognized in the total number of positive cases.
- F-Measure – The harmonic means of precision and recall. it is the important measure as it gives equal importance to precision and recall.

F – Measure = 2 x recall x precision / (precision – recall)

Table 2: Comparison results of Various Decision Tree Models with Referred Study

Study Decision Tree	Related Studies	TP	FP	Prediction	Recall	F-Measure	ROC – Curve Area	CLASS	Time Taken (sec)
J48	Study Done by Purwa Sewaiwar and Kamal Kant Verma	1	0	1	1	1	1	Y	0.14
		1	0	1	1	1	1	N	
Random Forest		0.838	0.014	0.969	0.838	0.899	0.964	Y	0.07
		0.986	0.016	0.924	0.924	0.954	0.962	N	
Random Tree		0.838	0.014	0.969	0.838	0.899	0.976	Y	0.08
		0.986	0.162	0.924	0.986	0.954	0.971	N	
LMT		1	0.014	0.974	1	0.987	1	Y	6.9
		0.986	0	1	0.986	0.993	0.99	N	
Decision stump		1	0	1	1	1	1	Y	0.18
		1	0	1	1	1	1	N	
Proposed Decision Tree	Proposed Study	0.638	0.014	0.769	0.738	0.799	0.776	Y	0.05
		0.686	0.016	0.724	0.786	0.754	0.771	Y	

Study done by Purwa Sewaiwar and Kamal Kant Verma " A Comparative Study of Various Decision Tree Classification Algorithm using WEKA", International Journal of Emerging Research in Management and Technology " , Volume-4 , June 10 is taken to compare the performance analysis of the proposed Rule Based Data Cleaning Algorithm using Decision Tree model.
In this comparison the results obtained were compared with the simulated results. Their study used student database for identifying the students qualifying for the degree and the proposed study

used Wollongong University Web Log server data to remove irrelevant data. In this case the tuples satisfying the rules are classified. The classified data are bad and non-classified are good data.
The referred study used WEKA tool to evaluate the performance of various Tree models and the proposed study uses MATLAB (7.9.0.529). Windows 8 (64 bit) was used as the platform to install Mat lab to develop and execute the applications. Intel (R) Celeron ® processor with 1.50 GHz clock speed and 2.00 GB RAM were the system configuration used to implement the proposed model.
From the results obtained from both the studies, it can be visualized the integrated behavior of the studies. In the referred study there is a contradictory in classifying results for most of the tree models with the proposed and simulated data, though the performance of Random Forest, random tree and LMT models results show better results.
But the proposed Decision Tree model's results align with the simulated model and the time taken to classify is less than the other models. Both the proposed and simulated classifying results are same. These observations prove the better and improved performance of the proposed decision Tree model over other related and referred Decision Tree models.

5.1.3. Comparison results of previous hashing functions with new hashing function with a new error factor

Table 3: Comparison Results of Previous Hashing Functions with New Hashing Function with a New Error Factor

Searching Techniques Used	Data Source Used / No of Records	Searching time in seconds	Big O Comparisons
Linear Search	Nehru Arts and Science College Web Log Server 10,000000	12.25	O(n)
Binary Search		11.25	O(log n)
Proposed hashing function version 1.0		10.48	O(log n)
Modified hashing Function version 2.0		9.38	O(log n)
New hashing function with error factor version 3.0		8.26	O(log n)
Linear Search	Murdoch University 10,000000	12.75	O(n)
Binary Search		11.56	O(log n)
Proposed hashing function version 1.0		10.01	O(log n)
Modified hashing function version 2.0		9.28	O(log n)
New hashing function with error factor version 3.0		8.36	O(log n)
Linear Search	ECMIT 10,000000	12.75	O(n)
Binary Search		11.56	O(log n)
Proposed hashing Function version 1.0		10.01	O(log n)
Modified hashing function version 2.0		9.01	O(log n)
New hashing function with error factor version 3.0		8.58	O(log n)

The results populated in the above table clearly prove the improved performance of the new hashing function with a new error factor over other popular searching techniques. The new hashing function has consumed 8.26 seconds for Nehru Arts and Science College web log data, 8.36 seconds for Murdoch University Web log data and 8.58 seconds for ECMIT web log data to search and locate the user record. These results are a strong evidence for the improved performance in accuracy and efficiency of the new hashing function with the new error factor.

5.1.4. Overall competitive analysis of the modified data cleaning and unique user identification algorithms with previous study for murdoch university

Table 4: Overall Competitive Analysis of the Modified Data Cleaning and Unique User Identification Algorithms with Previous Studies for Murdoch University

Data Sources	Murdoch University Version 1.0	Version 2.0	Version 3.0
Entries in Raw Web Log File	100000279900 (records)	100000279900 (records)	100000279900 (records)
Entries after Data Cleaning using proposed Data Cleaning Algorithm	100000002783 (records)	100000001883(records)	99999995883(records)
Number of Users	567502876	567502876	567502876
Number of Unique Users	43,5785	56,3467	57.2872
Execution time of proposed UUI Algorithm	3.2579(s)	2.5043(s)	2.2247(s)
Number of Sessions.	546744372	586744372	606526587

From the observations displayed in table 4, the modified algorithms show better results over the previous studies. On an average around two lakhs irrelevant records have been removed from the raw Web Log server by the modified rule based data cleaning algorithm. The increase in the number of users and unique users further add strong proof for the improved performance of the new UUI algorithm. Above all the execution time of the modified UUI algorithm to locate a new user has decreased from 2.5 to 2.22 seconds, which is a resilient evidence for the algorithms improved performance.

5.1.5. Overall comparative analysis of the modified data cleaning and unique user identification algorithms with previous study for ECMIT

Table 5: Overall Competitive Analyses of the Modified Data Cleaning and Unique User Identification Algorithms with Previous Studies for ECMIT

Data Sources	Emirates College of Management and Information Technology.		
	Version 1.0	Version 2.0	Version 3.0
Entries in Raw Web Log File	100450279900 (records)	100450279900 (records)	100450279900 (records)
Entries after Data Cleaning using proposed Data Cleaning Algorithm	100270002783 (records)	100270001983 (records)	100269989983 (records)
Number of Users	606920287	606920287	606920287
Number of Unique Users	56,3467	57,2878	57,3878
Execution time of proposed UUI Algorithm	4.437(s)	3.436(s)	2.85(s)
Number of Sessions.	546744372	586744372	604528176

From the results displayed in the above table 6 it is clearly proved the improved and better performance of the modified data cleaning and UUI algorithms respectively. Around twelve thousand irrelevant records have been removed from the previous study. And there are thousand new users more than the previous study. Similarly the execution time of the new UUI algorithm is apparently less over the previously proposed UUI algorithms. These facts add a strong evidence for the improved performance of the modified algorithms for Murdoch University Web Log data.

5.1.6. Overall comparison summary of modified data cleaning and unique user identification algorithm with previous study for murdoch university

From the results displayed in the above table 6, it is palpable that the results produced by the new modified data cleaning and UUI algorithms are better over the results of the previous algorithms. Increase in duplicate records, decrease in number of unique users and decrease in the overall bandwidth are the evidences for these facts. Still the study is focusing to improve the overall performances of the data cleaning and unique user identification algorithms in terms of accuracy and efficiency.

Table 6: Overall Comparative Summaries for Murdoch University

Month	No of Records	Unique Visitors	No of Duplicate Records	Number of Visits	Pages	Hits	Bandwidth (GB)	Unique Visitors	No of Duplicate Records	Number of Visits	Pages	Hits	Bandwidth (GB)
	Previous Study Version 2.0							Modified Study Version 3.0					
Jan 2014	90895423	83895423	84789231	7206192	8275923719	184759237	1.453	83883923	7196192	7206192	8275923719	184759237	1.256
Feb 2014	90905423	84532723	84664732	7240691	80652734	20365273	1.345	84522482	7228191	7240691	80652734	20365273	1.167
Mar 2014	90917423	85884575	85934575	5992848	74894572	16189457	1.445	85872575	5979598	5992848	74894572	16189457	1.236
Apr 2014	90931423	84635433	84777433	7173990	76765438	18176438	1.472	84621983	7164190	7173990	76765438	18176438	1.243
May 2014	90943423	83447893	83570893	8390530	75567899	18556899	1.565	83433604	8380765	8390530	75567899	18556899	1.347
June 2014	90953423	84109875	84339875	6643548	73329876	17432876	1.723	84095477	6632548	6643548	73329876	17432876	1.525
Total	454651115	101288591	25257500	3544607	381210519	57821943	9.003	42254121	3538292	3544607	381210519	57821943	7.774

6. Graphical results

Figure 6 displays the overall performance of proposed data cleaning algorithms version one, two and three for Nehru Arts and Science, Murdoch University and ECMIT. From the figures it is evident that more number of duplicate and irrelevant data is removed by data cleaning algorithm version three over other versions. Inclusion of decision tree classification model approach to make complex decisions and find an appropriate solution has improved the efficiency of the modified data cleaning algorithm over the previous versions.

Figure 7 illustrates the efficiency measures of various decisions tree models. ECMIT web log server data is taken for evaluation. A sample of ten thousand records is used to evaluate the models. Modified data cleaning algorithm with decision tree classification model approach has consumed just 0.05 seconds to classify irrelevant records. This result is better over other tree models. Performance of random forest tree model is equivalently good over the proposed data cleaning algorithm with decision tree classification model approach.

Modified unique user identification algorithm has just taken an average of 2.5 seconds to locate an existing web user from a record size of three billion records. The time taken is apparently less over the searching time of previous versions of proposed unique user identification algorithms.

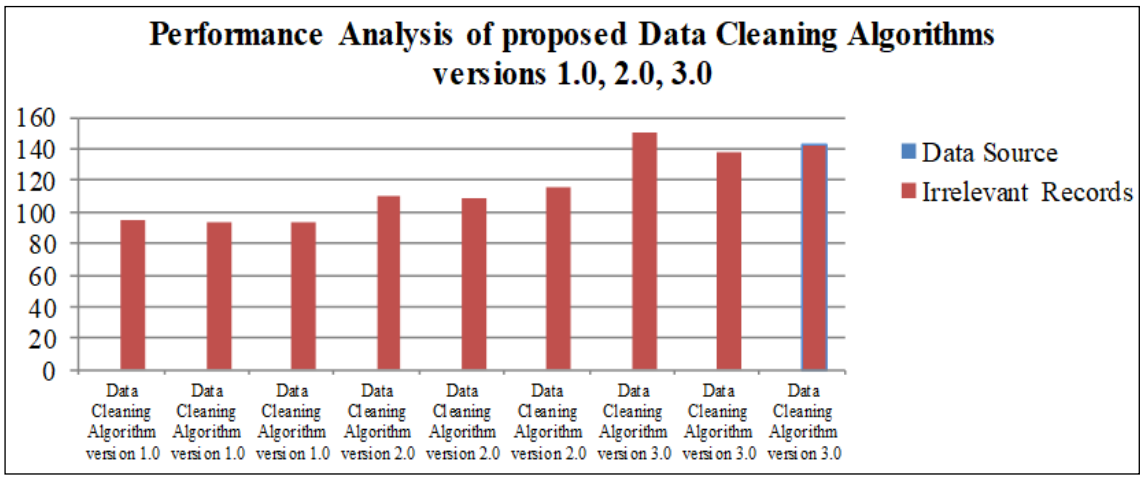

Fig. 6: Performance Analysis of Proposed Data Cleaning Algorithms Versions 1.0, 2.0, 3.0.

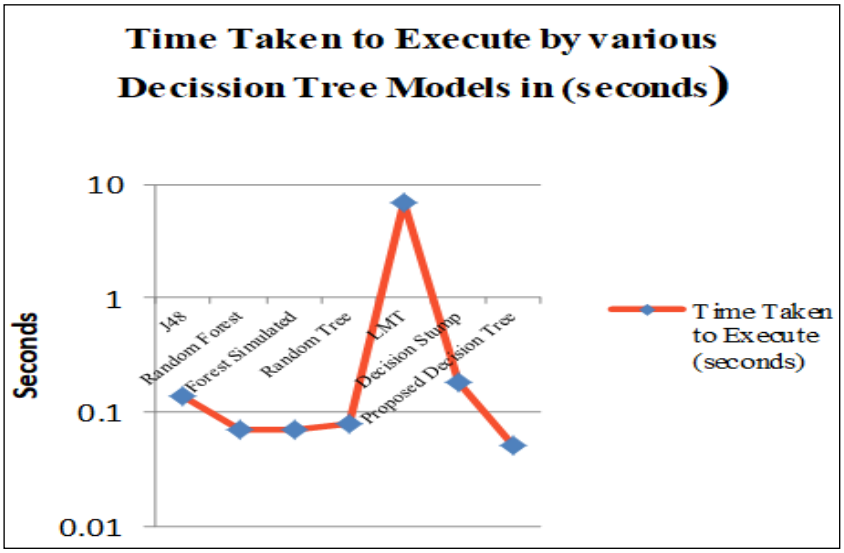

Fig. 7: Efficiency Measures of Various Decision Tree Models.

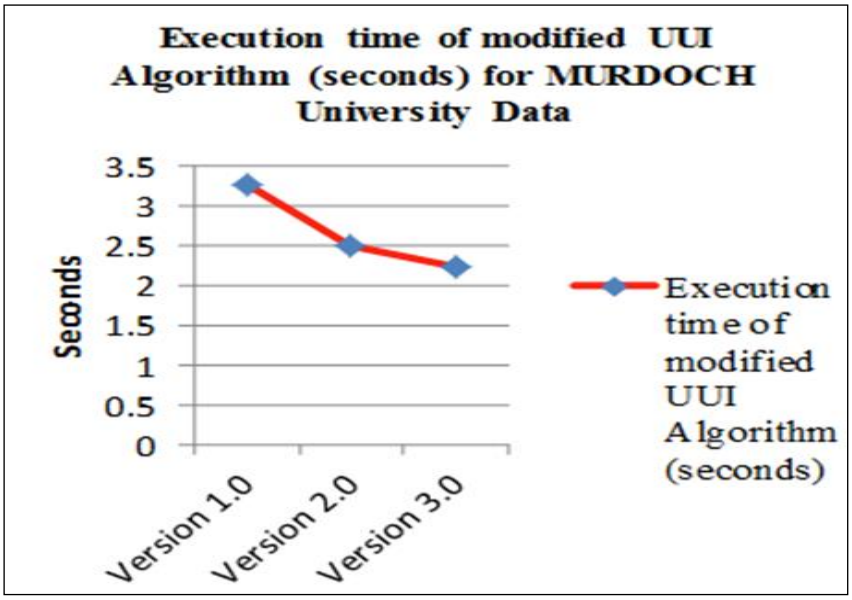

Fig. 8: Execution Time for UUI Algorithm Version 3.0 for Murdoch University Web Log Data.

Figure 9 displays the overall bandwidth occupied by the system to execute the suggested unique user identification algorithms. From the results depicted in the figure it is palpable that the system consumes less bandwidth for version 3.0 over version 2.0. The inclusion of decision tree based data cleaning and modified hashing techniques and hashing function along with binary search techniques helped to attain this milestone. Still studies are in progress to improve the efficiency and accuracy of the proposed unique user identification algorithm.

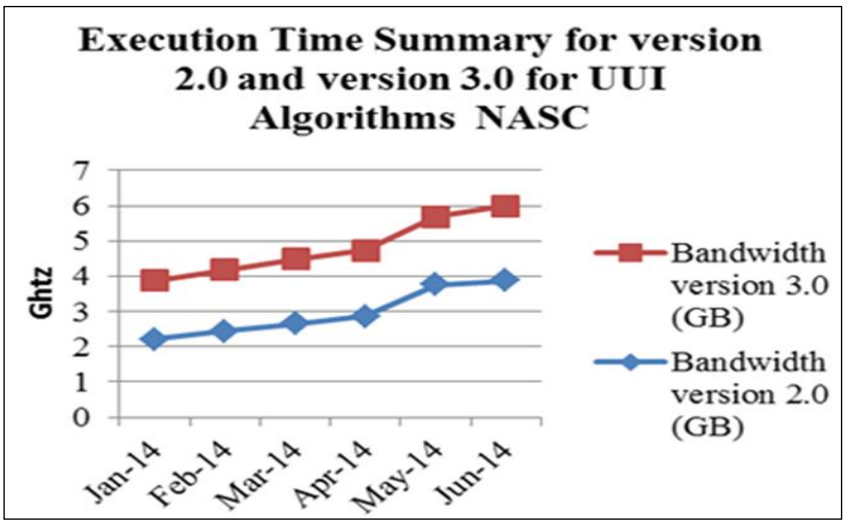

Fig. 9: Execution Time Summary for Version 2.0 and Version 3.0 of UUI Algorithms for NASC.

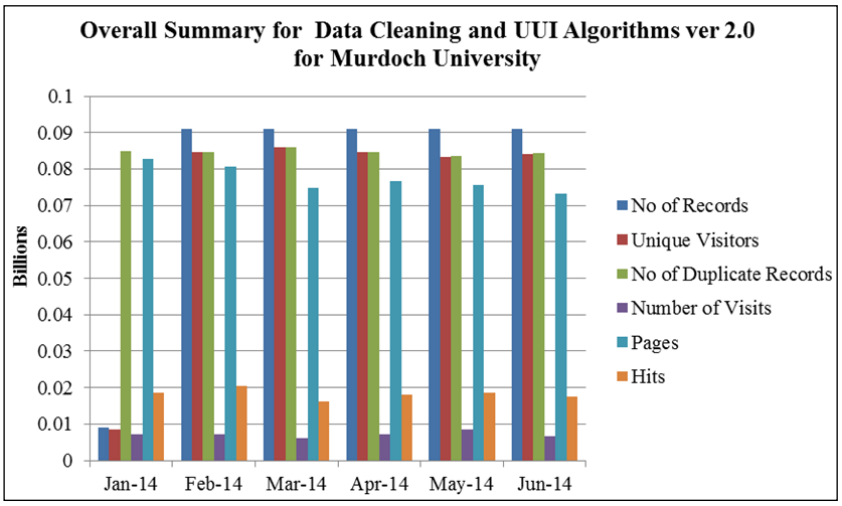

Fig. 10: (A): Overall Summary for Data Cleaning and UUI Algorithms Version 2.0 for Murdoch University.

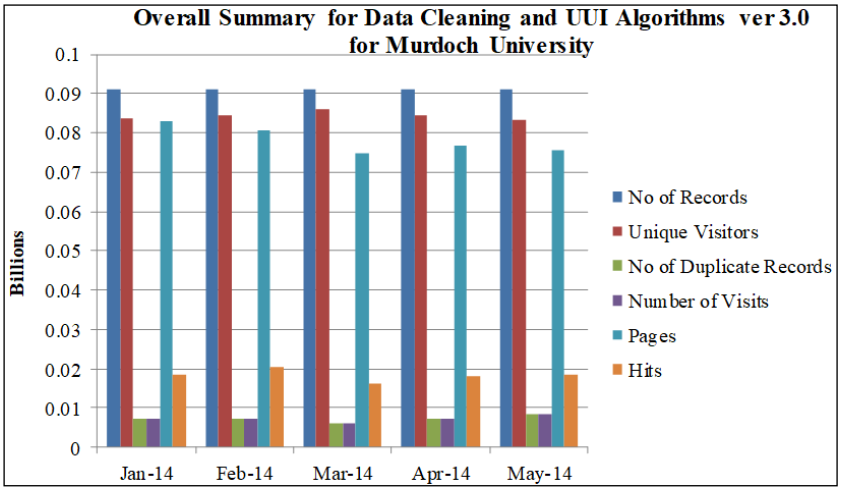

Fig. 10: (B): Overall Summary for Data Cleaning and UUI Algorithms Version 3.0 for Murdoch University.

Figures 10(a) and 10(b) depicts the overall performance of the suggested unique user identifications algorithm version 2.0 and version 3.0 from January 2014 till June 2014. From the figures it is proved that the performance of version 3.0 UUI algorithms is better over version 2.0. Increase in count of irrelevant data, increase in number of unique users, increase in number of sessions, increase in number of visits by users are various factors which contribute to the improved performance of version 3.0 UUI algorithm over version 2.0 UUI algorithm.

Introduction of decision tree classification model based data cleaning algorithm, modified hashing techniques and modified hashing function with a new error factor integrated binary search techniques in the suggested unique user identification algorithm have farther elevated the progress of this study.

7. Conclusion

This study is focusing on preprocessing techniques in web mining. Considering the importance of relevant and meaningful data, the study has proposed an innovative data cleaning algorithm and to understand the behavior of the organizations web sites, a unique

user identification algorithm is suggested and implemented in previous study. Further to improve the performance of the proposed algorithms this paper has introduced decision tree classification model approach to the earlier proposed data cleaning algorithm to simplify the rule framing and complexity in making decisions to remove irrelevant data. In addition the study has also introduced a modified hashing function with a new error factor. Modified hashing function along with binary search techniques have reduced the searching time of existing web user in web log server using the modified unique user identification algorithms. Various experimental analysis and report summaries generated prove the improved performance of the modified algorithms.

References

[1] Chitraa V & Dr.Antony Selvadoss, "A Novel Technique for Sessions Identification in Web Usage Mining Preprocessing", International Journal of Computer Applications, Vol.34, No.9, (2011), pp.23-31.

[2] Suguna R & Sharmila D, "User Interest Level Based Preprocessing Algorithms Using Web Usage Mining", International Journal on Computer Science and Engineering, Vol.10, (2015), pp.108-117.

[3] Vaarandi R & Pihelgas M, "Logcluster-a data clustering and pattern mining algorithm for event logs", 11th International Conference on Network and Service Management, (2015), pp.1-7. https://doi.org/10.1109/CNSM.2015.7367331.

[4] Jagan S & Rajagopalan SP, "A Survey on Web Personalization of Web Usage Mining", International Research Journal of Engineering and Technology, Vol.02, No.01, (2015), pp.2395-0056.

[5] Parmar VP & Kumbharana CK, "Comparing Linear Search and Binary Search Algorithms to Search an Element from a Linear List Implemented through Static Array, Dynamic Array and Linked List", International Journal of Computer Applications, Vol.121, No.3, (2015).

[6] Ranjena Sriram & Mallika R, "Innovative Pre-Processing Technique and Efficient User Identification Algorithm for Web Usage Mining", International Journal of Advanced Research in Computer Science and Software Engineering, Vol.6, No.2, (2016), pp.85-90.

[7] Sleator DD & Tarjan RE, "Self-adjusting binary search trees", Journal of the ACM (JACM), Vol.32, No.3, (1985), pp.652-686. https://doi.org/10.1145/3828.3835.

[8] Singh K & Sulekh R., "The Comparison of Various Decision Tree Algorithms for Data Analysis", International Journal of Engineering and Computer Science, Vol.6, No.6, (2017).

[9] Sewaiwar P & Verma KK, "Comparative Study of Various Decision Tree Classification Algorithm Using WEKA", International Journal of Emerging Research in Management &Technology, Vol.4, (2015), pp.2278-9359.

[10] Chourasia S, "Survey paper on improved methods of ID3 decision tree classification", International Journal of Scientific and Research Publications, Vol.3, No.12, (2013).

[11] Vadhera P & Lall B, "Review Paper on Secure Hashing Algorithm and Its Variants", International Journal of Science and Research (IJSR), Vol.3, No.6, (2012), pp.55-61.

[12] Raiyani SA, "Preprocessing and Analysis of Web Server Logs", International Journal of Computer Science & Communication Networks, Vol.2, (2015), pp.46-55.

[13] Suneetha KR & Krishnamoorthi R, "Identifying User Behavior by Analyzing Web Server Access Log File", International Journal of Computer Science and Network Security, Vol.9, No.4, (2009), pp.327-332.

[14] Sahu MS & Sahu APL, "A Survey on Frequent Web Page Mining with Improving Data Quality of Log Cleaner", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol.4, No.3, (2015), pp.825-829.