ISSN : 2455-9679
Impact Factor : 2.015

IJRTSM

INTERNATIONAL JOURNAL OF RECENT TECHNOLOGY SCIENCE & MANAGEMENT

"ADVANCEMENT OF DECISION TREE ANALYSIS FOR MINING WEB DATA STREAM"

***Neha Sahu*[1], *Pritesh Jain*[2]**

[1] *Research Scholar, Dept. Computer Science & Engineering PCST Indore ,(M.P.), India*

[2]*Assistant Professor, Dept. Computer Science & Engineering, PCST Indore,(M.P.) India*

ABSTRACT

Two important challenge included with web use mining are processing the raw data to give a close to reality or genuine number image of how site is being utilized, and filtering the outcome of various data mining set of computer guidelines in order to present only rules and patterns. In this work we create decision tree analysis, which is proficient mining strategy to mine log files and extract knowledge from web data stream and generated training rules also, Pattern which are useful to discover distinctive data which will be in relation with log file. The issue to extract knowledge from substantial raw data has developed as another data structure. Data stream is another period in the field of data mining. Various calculations are utilized for preparing and grouping data streams. Old calculations are most certainly not suitable to process data stream which in cause produce issues with respect to arrangement. A process which is created from stream data for grouping must refresh incrementally after the new entry of latest data. Data stream classification performance can be observed by different parameters, for example, accuracy, computational speed, memory and time taken for execution. Data stream order analysis must need to meet some necessities and measures to deal with nonstop flow of data. These analysis get less time traverse to assess information and build demonstrate, might be just once with less measure of resource, time and prediction. So to think about the ideas of classification algorithms will lead towards advancement of better approaches for stream data mining. In our thesis work we create decision tree analysis, which is proficient mining strategy to mine log files and extract knowledge from web data stream and produced training rules which are useful to discover diverse data with respect to log file.

KEYWORDS : *Web Usage Mining, Decision Tree, Temporal Rule Mining, Data Stream.*

I. INTRODUCTION

Web Usage Mining

Predicated on the distinctive highlight and diverse approaches to acquire information, web mining can be separated into two noteworthy components: Web Contents Mining and Web Utilization Mining. Web Contents Mining can be characterized as the automatic search and retrieval of information and subsidiary things/valuable supplies accessible from a huge number of sites and on-line (computer files loaded with information) through search engines / web spiders. While; Web Utilization Mining can be depicted as the disclosure and investigation of utilizer access pattern, through the mining of log documents and associated information from a specific Web webpage. Web use mining is utilization of data mining methods for getting things done to web click stream data for extracting usage data. As website keep on growth in size and complex trouble, the outcome of web usage mining have turned out to be fundamental for some application, for example, site design, two important challenge engaged with web usage mining are processing the raw data to give a (near reality or

ISSN : 2455-9679
Impact Factor : 2.015

genuine number) picture of how website is being utilized, and filtering the outcome of various data mining set of computer directions so as to introduce just protocols and patterns.

Web usage mining, from the data mining perspective, is the assignment of applying data mining procedures to find usage patterns from Web data keeping in mind the end goal to understand and better serve the necessities of clients exploring on the Web each data mining task, the procedure of Web usage mining additionally comprises of three fundamental steps: (I) preprocessing, (ii) pattern discovery and (iii) pattern analysis. In this work pattern discovery implies applying the introduced frequent pattern discovery strategies to the log data. Hence the data must be converted over in the preprocessing stage with the end goal that the outcome of the conversion can be utilized as the input of the algorithms. Pattern analysis implies understanding the outcomes got by the algorithms and drawing conclusions.

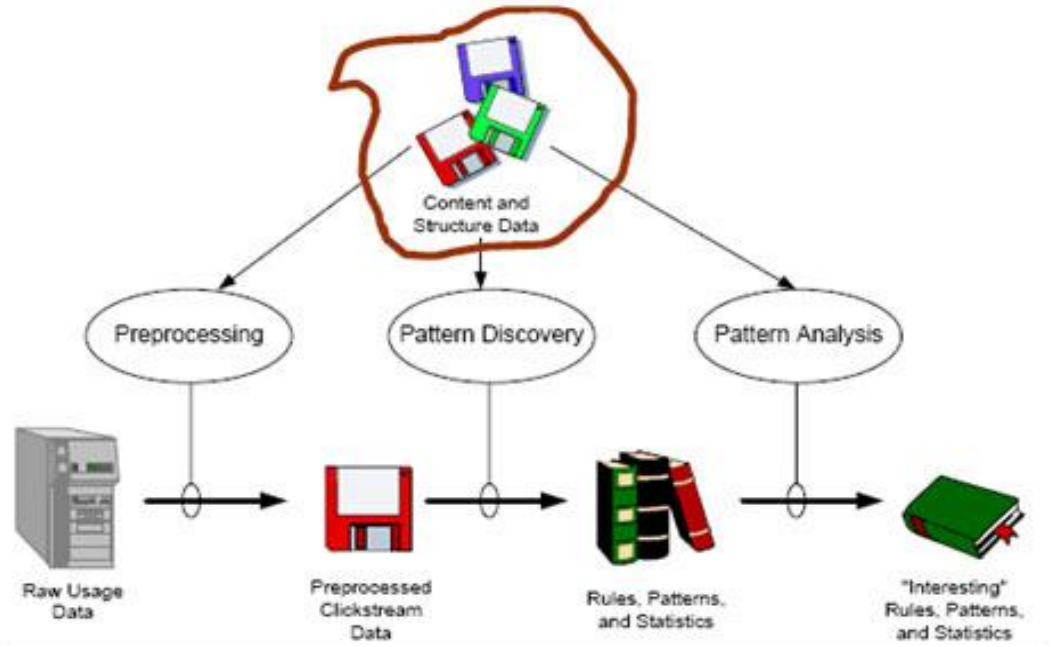

Figure 1.1 Web Usage Mining Process

Figure shows the methodology of Web usage mining recognized as a contextual analysis in this work. As can be seen, the information of the procedure is the log information. The data must be preprocessed with a particular ultimate objective to have the best possible information for the mining calculations. The differing routines require distinctive information positions, in this way the preprocessing stage can give three sorts of yield data. The successive examples disclosure stage needs only the Web pages passed by a given customer. For this circumstance the arrangement of the pages are discarded. Furthermore the copies of similar pages are discarded, and the pages are requested for in a predefined request.

Web use mining is utilization of data mining methods for getting things done to web click stream data for extracting usage data. As website keep on growth in size and complex trouble, the outcome of web usage mining have turned out to be fundamental for some application, for example, site design, two important challenge engaged with web usage mining are processing the raw data to give a (near reality or genuine number) picture of how website is being utilized, and filtering the outcome of various data mining set of computer directions so as to introduce just protocols and patterns.

Plenty of research is being carried out and still continuously going on for this work, here we are going to develop decision tree algorithm, which is efficient mining technique to mine log files and extract information from web data stream and developed training rules and Pattern which are helpful to find out different information related to log file. We increase accuracy of generating non redundant association rule for both nominal and numerical data with less time complexity and

memory space. In this method we use N-fold cross validation technique for performance evaluation and for classification of data set we are use decision learning algorithm with some modification in decision tree algorithm.
To resolve the need of effective and efficient algorithm we propose our solution based on following facts:

1. Search a most frequently used sequential web log mining algorithm
2. Implement the found algorithm
3. Find the performance study of that algorithm.
4. Compare the performance parameters for comparative analysis

Decision tree learning is a method commonly used in data mining. Decision tree induction is the learning of flowchart-like tree structure from class-labelled training examples. A decision tree has three principle parts: hubs, leaves, and edges. Every hub is marked with a trait by which the information is to be parcelled. Every hub has various edges, which are named by estimations of the quality. An edge unites either two hubs or a hub and a leaf. Leaves are named with a choice quality for classification of the information. To settle on a choice utilizing a decision Tree, begin at the root hub and take after the tree down the branches until a leaf hub speaking to the class is come to. Every decision tree speaks to a tenet set, which classifies information as per the qualities of dataset. The decision tree building calculations might at first forms the tree and afterward prune it for more viable arrangement. With pruning strategy, segments of the tree may be uprooted or consolidated to lessen the general size of the tree. The time and space many-sided quality of developing a decision tree relies on upon the measure of the information set, the quantity of traits in the information set, and the state of the subsequent tree. The development of decision tree classifiers does not require any space information, and can deal with high dimensional information. The learning and characterization procedure of decision tree impelling is basic and quick.

II. LITERATURE REVIEW

[1] **Nazli Mohd Khairudin, Aida Mustapha, and Mohd Hanif Ahmad,2014,** Proposed the advent of web-based applications and services has created such diverse and voluminous web log data stored in web servers, proxy servers, client machines, or organizational databases. This paper attempts to investigate the effect of temporal attribute in relational rule mining for web log data. We incorporated the characteristics of time in the rule mining process and analysed the effect of various temporal parameters. The rules generated from temporal relational rule mining are then compared against the rules generated from the classical rule mining approach such as the Apriori and FP-Growth algorithms. The results showed that by incorporating the temporal attribute via time, the number of rules generated is subsequently smaller but is comparable in terms of quality.

[2] **Anil Agrawal , Mohd. Husain , Raj Gaurang Tiwari , Suneel Vishwakarma, 2011,** propose new approaches for database selection and documents selection. In the first part of our work we present an algorithm DBSEL for database selection. This algorithm selects those databases from no. of databases which contain query ‘q’. This algorithm test each database with its documents stored in it. If any document of database contains the query ‘q’ at least one time then we select that database. If all the documents of database does not contains the query ‘q’ then that database will not be selected. In the second part of our work we present an algorithm HighRelDoc for documents selection. This algorithm search all the selected databases and select only those documents from each database in which the query ‘q’ occurs at least one time. After that this algorithm ranks all the selected documents according to the no. of occurrence of query ‘q’ in descending order. Finally this algorithm returns the top ‘n’ most relevant documents from the sorted list of documents for any positive integer ‘n’.

[3] **D. Vasumathi & Dr. A Govardhan, 2009,** propose a novel FBCA approach for web usage mining. In our approach, the FBCA technique is applied to mine association rules from web usage lattice constructed from web logs. The discovered knowledge(association rules) can then be used for practical web applications such as web recommendation and personalization. We apply the FBCA-mined association rules to web recommendation and compare its performance with that of classical Apriority -mined rules. The results indicate that the proposed FBCA approach not only generates far

fewer rules than Apriority-based algorithms, the generated rules are also of comparable quality with respect to three objective performance measures.

[4] **Chungsheng Zhang & Liyan Zhuang, 2008,** The article discusses the importance of data preprocessing in web mining and gives the topology structure for the website in the view of actual condition, analyzes the limitation of reference [3] and proposes a data structure based on adjacency list. The proposed method satisfies the actual condition of topology structure for the existed website. The special data structure and path filling algorithm based on adjacency list are given. The data structure satisfies the commonness of topology structure for the existed website and the time complexity is lower.

[5] **Jungie Chen and Wei Liu(2006),** Web usage mining is a kind of data mining that it mines the information of server logs after users browse the Web pages. After introducing the staple technology and process of Web usage mining, the article advances an architecture of Web usage mining and presents the work principle of system. In addition, the article discusses some key technologies in system design such as session identification and data cleaning at length.

III. METHODOLOGY

Decision Learning Technique

Decision Tree Learning is a strategy normally utilized as a part of data mining. Decision tree acceptance is the learning of flowchart-like tree structure from class-labeled preparing illustrations. A decision tree has three standard parts: hubs, leaves, and edges. Every hub is marked with a trait by which the information is to be parceled. Every hub has various edges, which are named by estimations of the quality. An edge unites either two hubs or a hub and a leaf. Leaves are named with a choice quality for classification of the information. To settle on a choice utilizing a decision Tree, begin at the root hub and take after the tree down the branches until a leaf hub speaking to the class is come to. Every decision tree speaks to a tenet set, which classifies information as per the qualities of dataset. The decision tree building calculations might at first forms the tree and afterward prune it for more viable arrangement. With pruning strategy, segments of the tree may be uprooted or consolidated to lessen the general size of the tree. The time and space many-sided quality of developing a decision tree relies on upon the measure of the information set, the quantity of traits in the information set, and the state of the subsequent tree. The development of decision tree classifiers does not require any space information, and can deal with high dimensional information. The learning and characterization procedure of decision tree impelling is basic and quick.

A decision tree is essentially a stream chart of inquiries or data directs that finally leads toward a decision. For example, an auto buying decision tree may start by asking whether you require a 1999 or 2000 model year auto, at that point request what sort from auto, at that point ask whether you incline towards force or economy, and whatnot. Finally it can make sense of what might be the best auto for you.

Decision trees structures are participated in item choice frameworks offered by various traders. They are amazing for conditions in which a visitor goes to a Web site with a particular need. However, once the choice has been made, the responses to the inquiries contribute little to focusing on or personalization of that guest later on.

N-Fold Cross validation

Cross-acceptance, in some cases called revolution estimation, is a procedure for surveying how the after effects of a measurable investigation will sum up to an autonomous information set. It is for the most part utilized as a part of settings where the objective is forecast, and one needs to appraise how precisely a prescient model will perform practically speaking. One round of cross approval includes dividing a specimen of information into integral subsets, performing the investigation on one subset (called the preparation set), and accepting the examination on the other subset (called the approval set or testing set). To diminish variability, various rounds of cross-acceptance are performed utilizing distinctive

ISSN : 2455-9679
Impact Factor : 2.015

parcels, and the approval results are found the middle value of over the rounds. Cross-acceptance is imperative in guarding against testing theories recommended by the information particularly where further specimens are unsafe, unreasonable or difficult to gather.

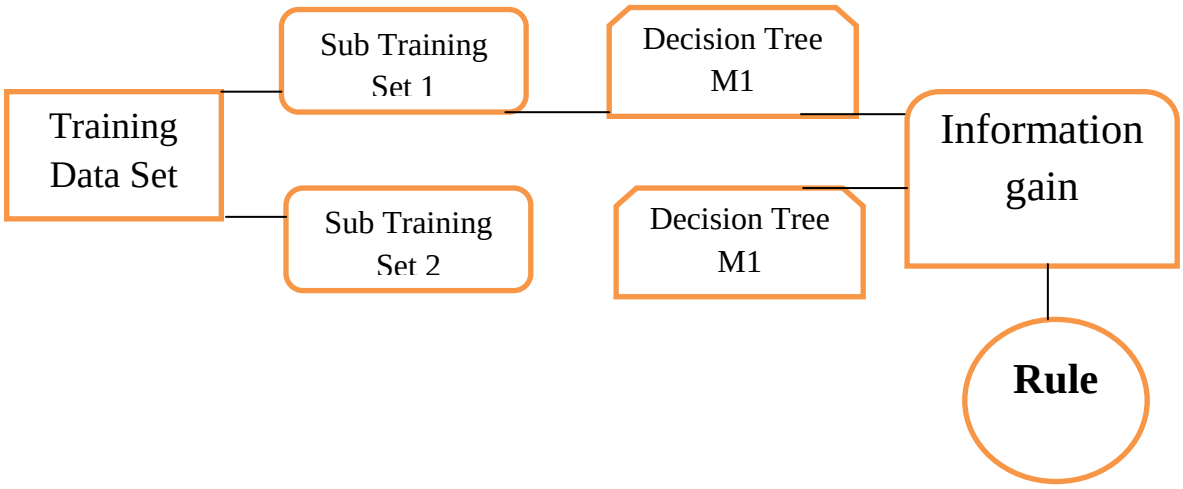

Figure 3.1 Decision Tree Classifiers

Our undertaking is planned with the fundamental goal to mine log fils and extract knowledge from the experimental web log and after the generation of training rules these rules are useful to discover diverse data related with log file. For that reason we propose architecture to produce the rules from the experimental data collection. This is done in these stages

1. Data accumulation
2. Data preparing utilizing chose display
3. Model building and model assessment
4. Pattern Discovery

The proposed work and thesis follows the following steps:

The proposed work and proposition takes after the accompanying advances:

1. Experimental data selection: In this stage needed to input log files in to the system for analysis the input log files are in w3c arrange.
2. Data processing: In this stage system will clean the information and separate them and perform their arrangement.
3. Model building and evaluation: In this stage of stage processing using the supplied data is changed over in to data model utilizing the choice of algorithm other words selected data model is utilized to set up a navigational model for queries of client.
4. Performance investigation: in this stage we compute the execution parameters for comes about examination.

ISSN : 2455-9679
Impact Factor : 2.015

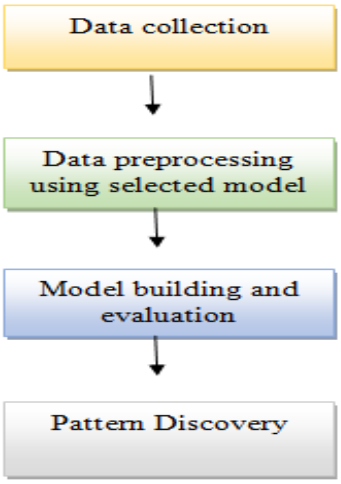

Figure 3.2 The basic structure of our proposed model

Proposed System Architecture

Below given diagram shows the system architecture of desired system. In this diagram we show the different sub systems of the complete system. These sub systems are work together and form the complete system. To describe complete systems working we describe each stage of processing one by one.

- Experimental data selection: using this module we supply input to the system and using this data we prepare navigational model in next phase.
- Algorithm selection: here required to select an appropriate data model to work with.
- After selection of algorithm there are two different algorithms are implemented and using the data we generate data model according to the supplied data.
- Model generation: selected algorithm here works over the supplied data and generates rules for prediction.
- Result evaluation: after model generation here we check the authenticity of models and evaluate performance parameters.

IV. IMPLEMENTATION

Algorithm Used

1. select data set D
2. find list of all attributes in data set
3. check attributes data types
4. if all attributes = numerical data type
    a. get average of each attributes mark as threshold value
    b. compare with all selected attributes
        i. if attribute value <= threshold then
        ii. mark as 0
        iii. else
        iv. mark as 1
        v. end if
    c. find distance from all instance of data set
    d. arrange according to distance
5. if all attributes = nominal data type then

ISSN : 2455-9679
Impact Factor : 2.015

a. find all unique attributes to attributes list
b. get threshold for each attributes using the given formula
c. threshold = (total unique values/ total count)$\log_n$ (total unique values/ total count)
d. calculate the index of each unique value using the given formula=
(no of values in list/ total values)$\log_n$(no of values in list/ total values)
e. Assign label index to the values and compare with threshold
f. Find distance for all instance

6. end if
7. return classes

V. RESULTS

(a) Accuracy: accuracy of the system is defined by the actually predicted values verses wrong values predicted. The accuracy of system is calculated using the cross validation in this method we calculate the values using given formula

$$Accuracy = \frac{total\ values - wrong\ values}{total\ values} X\ 100$$

Accuracy of the system is derived using above formula and here we include the results obtained by the system in first five experiments

Table 5.1 Comparison of Accuracy of both Temporal Mining and New Algorithm

No. of Execution	Temporal Mining	New Algorithm	No. of attribute taken
1	71.50 (support=2)	87.7%(No.of fold=2)	3
2	83.45%(support=3)	98.77%(No.of fold=3)	3
3	71.24%(support=4)	86.81%(No.of fold=4)	3
4	71.26%(support=5)	93.25%(No.of fold=5)	3
5	71.26%(support=5)	99.91%(No.of fold=5)	3

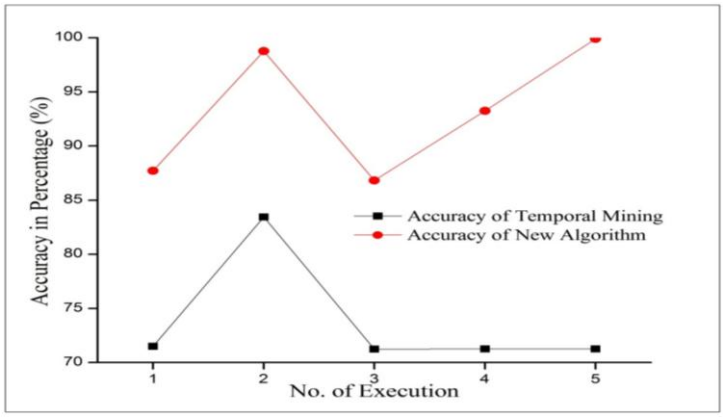

Figure 5.1 Graphical representation of Accuracy

The evaluation of results is performed for Temporal Mining by minimizing the support and increase the parameter after applies such condition we found that when as we minimize the support and increase the parameters accuracy of system decreases.

ISSN : 2455-9679
Impact Factor : 2.015

Moreover it proposed method include all parameters and thus simulate better results for the evaluation of such kind of data.

(b) Execution Time: to find the execution time we calculate the time required to build model results evaluation time included and we found that below given results.

Table 5.2 Comparison of Execution of both Temporal Mining and New Algorithm

No.of Execution	Temporal Mining	New Algorithm	No. of attribute taken
1	0.77 (support=2)	0.521 (No. of fold=2)	3
2	1.53 (support=3)	1.063 (No. of fold=3)	3
3	1.36 (support=4)	1.08(No. of fold=4)	3
4	1.25 (support=5)	1.10 (No. of fold=5)	3
5	2.17 (support=5)	1.94 (No. of fold=5)	3

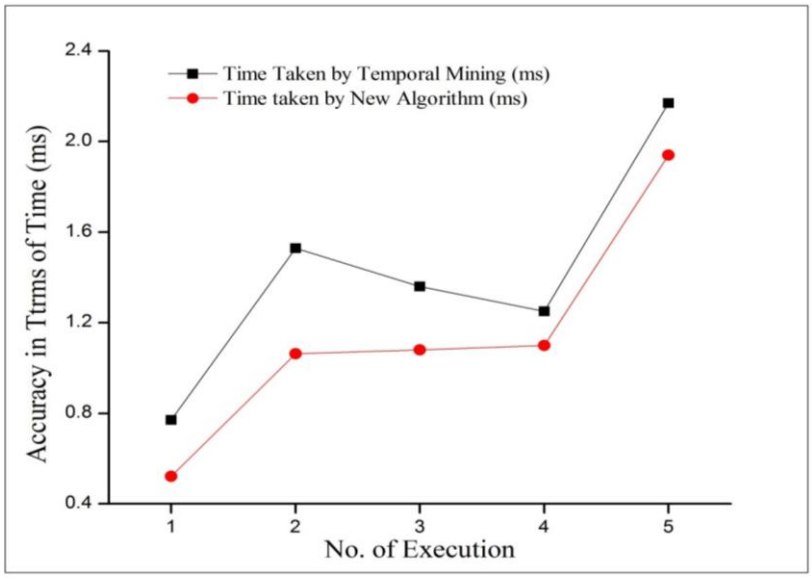

Figure 5.2 Graphical representation of Execution time

According to our analysis we found that execution time simulate is our algorithm is better than Temporal Mining Because the Temporal Mining time consumption graph is more uneven than proposed algorithm. And it is also considered that most of the time our model is much efficient then Temporal Mining.

(c) Memory uses: requirement of main memory to execute the algorithm is defined as memory uses. The results simulate the memory used in terms of MB.

Table 5.3 Comparison of Memory Consumption of both Temporal Mining and New Algorithm

No.of Executio	Temporal Mining	New Algorithm	No. of attribute taken

ISSN : 2455-9679
Impact Factor : 2.015

1	20.051 (support=2)	81.49(No. of fold=2)	3
2	85.74(support=3)	104.79 (No. of fold=3)	3
3	55.41 (support=4)	51.49(No. of fold=4)	3
4	16.82(support=5)	47.18(No. of fold=5)	3
5	98.52(support=5)	57.50(No. of fold=5)	3

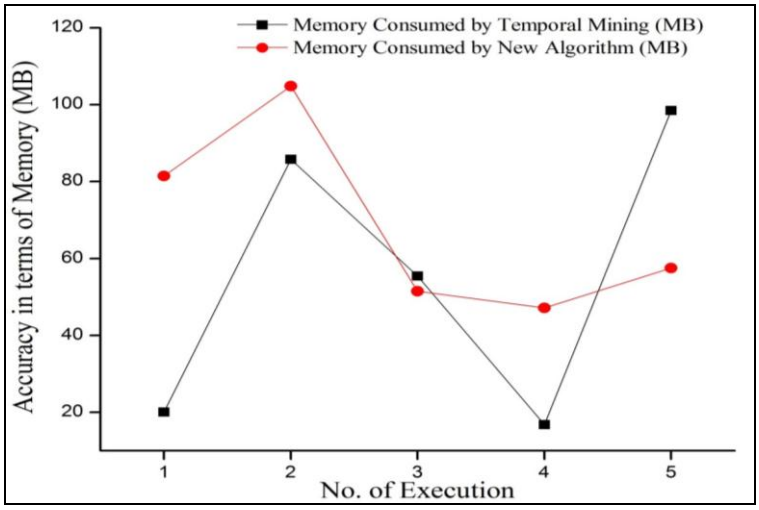

Figure 5.3 Graphical representation of Memory Consumption

Using above results we can clearly see that temporal rule algorithm consumes less memory then our proposed algorithm

VI. CONCLUSION

After evaluation of the obtained results, it was observed that the proposed work withstand with all the supplied input parameters. However the temporal Mining calculation resulted to work with selected parameters. Also, it was observed that the developed new algorithm performs better precise results then temporal mining although it was achieved by a fewer compromise of Memory employments. Consequently we can synopses the accompanying truths about our work. Accuracy of proposed algorithm 75%-85% is better than Temporal Mining algorithm. Memory uses of proposed algorithm were found to be higher than Apriori. Time required to execute model is 85%-95% less than Temporal Mining algorithm. Our proposed analysis is good analysis but when the situation where required less resource it will fail to work with low configuration system. Memory uses of proposed algorithm is 80%-85% is higher than Temporal Mining algorithm. Temporal mining performs better where the need of resources are less and supplied parameters are less.

REFERENCES

[1] Nazli Mohd Khairudin, Aida Mustapha, and Mohd Hanif Ahmad (2014). Effect of Temporal Relationships in Associative Rule Mining for Web Log Data System. Hindawi Publishing Corporation Scientific World Journal.

[2] Agrawal, M. Husain, R. G. Tiwari, and S. Vishwakarma(2011), "Web information recuperation from strewn text resource systems,"International Journal of Advances in Engineering and Technology,vol. 1, no. 2, pp. 126–137

ISSN : 2455-9679
Impact Factor : 2.015

[3] Arumugam G. and SugunaS(2009),"Optimal Algorithms for Generation of User Session Sequences Using Server Side Web User Logs, ",ESRGroups, France.

[4] D. Vasumathi and A. Govardhan(2009), "Efficient web usage mining based on formal concept analysis," Journal of Theoretical and Applied Information Technology, vol. 9, no. 2, pp. 99–109.

[5] Chungsheng Zhang and LiyanZhuang(2008) , "New Path Filling Method on Data Preprocessing in Web Mining ,", Computer and Information Science Journal.

[6] V. S. Tseng, K.W. Lin, and J.C. Chang(2008), "Prediction of user navigation patterns by mining the temporal web usage evolution," Soft Computing, vol. 12, no. 2, pp. 157–163

[7] Zhuang, L., Kou, Z., & Zhang, C. (2005). Session identification based on time interval in web log mining. In Intelligent information processing II (pp. 389-396): Springer-Verlag

[8] E. Winarko and J. F. Roddick(2007), "ARMADA—an algorithm for discovering richer relative temporal association rules from interval-based data," Data and Knowledge Engineering, vol. 63, no. 1, pp. 76–90 .

[9] E. Keogh, J. Lin, S.H. Lee (2007), and H. Van Herle, "Finding the most unusual time series subsequence: algorithms and applications," Knowledge and Information Systems, vol. 11, no. 1, pp. 1–27.