ISSN:0975-9646

An Enhanced Clustering Technique Using Rough Set Approach for Discovering Improved and Unambiguous Groups of Users

Parth Suthar[#1], Prof. Bhavesh Oza[*2]

[#]*M.E., Computer Science and Engineering, L. D. College of Engineering*
Ahmedabad, Gujarat, India

[*]*Assistant Professor, Computer Engineering Department, L. D. College of Engineering*
Ahmedabad, Gujarat, India

***Abstract*— Web is a very wide and well reached phenomenon. Its enormous popularity stems from the fact that it provides an enormous wealth of information on almost every conceivable subject. People use Web for vast amount of applications like online shopping, online bill payment, entertainment, social networks, education, marketing, data sharing, data storage etc. Due to these activities, web is flooded with large amount of data in the form of the access logs at web servers and proxy servers on daily basis. Leveraging this data and harnessing hidden information from it proves to be a very useful analysis task to improvise the services provided over the Web to better serve the users and also for the longer sustainability of the web itself. This analysis is called web usage mining which is a part of a broader concept called web mining which in turn, is part of data mining.**

The basic objective of this dissertation is to develop an improved clustering technique for grouping web users based on the similarity among their page access sequences. Expected results are accurate and hard groups of web users having maximum similarity among their page access sequences and thus, in their interests. These groups are used as training set for a classifier. Then, for every new user of the system, its cluster is predicted using the classifier. This study has its applications in areas like Web personalization, Site modification, Pre-fetching and caching, Market segmentation, Measurement of the returns of online advertising campaigns, Business Intelligence and many more.

***Keywords*— Web Usage Mining, Transactional data, Clustering, Prediction, Rough sets, K-NN, DBSCAN, Rough agglomerative clustering.**

I. INTRODUCTION

Clustering is widely used technique in data mining applications. It groups the objects based on similarities among them. To make recommendation effective, web users are compared with available information about their interactions with the system to find the similarity among them based on the pages they visit. The information about users' page visits are stored on servers and proxies in the form of web access logs. Web access data are dense and contains large amount of inconsistencies. It also contains the access information of a few web pages by plenty of users, thus ambiguous and making it hard to analyse accurately by normal clustering techniques. In general, well-developed clustering techniques use squared error as the similarity criterion among the data elements. This criterion is not able to capture the effects of ambiguity and uncertainty within the data. [4][5] As a result, the analysis of results generated by normal clustering techniques are inaccurate. [9][11]

Soft computing techniques should be used to deal with the ambiguity of the web access data. Rough set theory is one of the soft computing techniques which has a mathematical approach to uncertain and vague data. [17][10] It is able to represent a given set into two sets called lower approximations and upper approximations. The Lower approximation of given set represents the elements which are present in this set only and the upper approximation represents the elements that may present in other sets. [9][11][16][17] [19] Rough set theory presents a method to deal with inconsistent, ambiguous and noisy data. The similarity index used to compare the pair of user access sequences is Sequence and Set Similarity (S^3M) which is able to capture the constitution as well as the order of the pages within the sequences. The resultant sets after applying this method are unambiguous, certain and possible thus accurately analysable.

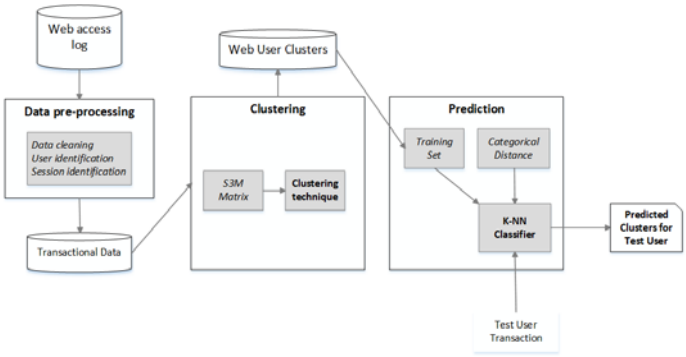

Figure 1 System Architecture

The system architecture is as shown in figure 1. It is based on the architecture given in [20]. To achieve the objective, the web access log files which contains the URLs is extracted and pre-processed. In pre-processing, the file is cleaned, formatted and grouped into meaningful transactions. These transactional data is then applied to clustering process. First here, the S3M similarity matrix, which is based on the rough set theory, is computed for the transactions. This matrix is used as distance measure with a

clustering technique. The output of this step is clusters of transactions being most similar to each other. These clusters are used as training set for a classifier called K- Nearest Neighbour. The classifier takes the unknown user transaction as input and predicts its cluster using the training clusters. K-Nearest Neighbour classifier uses the Euclidean distance for categorical attribute to find the distance of the unknown user transaction with the transactions present in the training clusters. The approach of prediction is the application of the clustering process.

Applying the rough set approach with an existing clustering technique DBSCAN can produce crisp and unambiguous groups of users based on the similarity among their interests and behaviours. Due to this, the resultant clusters are improved than the clusters discovered by previous techniques like rough agglomerative clustering, hence can be effectively applied to areas like web personalization, recommendation systems and market segmentation. [4]

II. BACKGROUND KNOWLEDGE

A. Rough set theory

The Rough Set theory was introduced by Zdzislaw Pawlak in the early 1980s, and deals with the classification analysis of data tables. Rough sets made it possible to develop efficient heuristics searching for relevant relations that allow to extract interesting patterns in data.

Rough set analysis facilitates approximation of concepts from the captured data. In this section, basics of rough set theory have been described. Let U be a universe, X be a set belonging to U and let R be an equivalence relation on U called an indiscernibility relation. The lower and upper approximations of X can be written as

$$\underline{R}(X) = \{x \in U, [x]_R \subseteq X\}$$
$$\overline{R}(x) = \{x \in U, [x]_R \cap X \neq \emptyset\}$$

Where $[x]_R$ denotes the equivalence class of the relation R containing the element x and R in the subscript denotes the family of all equivalence classes. X is called rough with respect to R iff $\underline{R}(X) \neq \overline{R}(X)$. Otherwise X is R-discernible. Rough set of X is defined by AR(X) = ($\overline{R}(X)$, $\underline{R}(X)$).

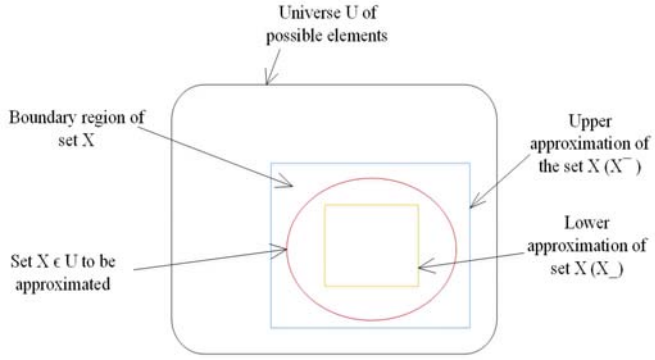

Figure 2 Rough Set and Approximations

The set BNR(X) = $\overline{R}(X)$ - R_X is called rough-boundary of X and the set U - R¯X is called negative region of X.

The Lower and upper approximations follow following properties [19]

- An object x can be part of at most one lower approximation. This implies that any two lower approximations do not overlap.
- If an object v is not part of any lower approximation it belongs to two or more upper approximations. This implies that an object cannot belong to only a single boundary region.

The rough set theory has following advantages.

- It proposes a method to handle inconsistent, ambiguous and noisy data.
- The resultant data objects are categorized into certain and possible sets.
- It has a clear mathematical background.

In general, the discovered knowledge or any unexpected rules are likely to be imprecise or incomplete, which requires a framework with soft computing techniques like rough sets.

B. S^3M

Given two transactions A and B, the measure of similarity between A and B is given by [9] [11] [16]

$$sim(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

This similarity measure is also called Jaccard similarity measure. [17] From the above definition it can be seen that sim (A, B) ∈ [0, 1]. sim(A, B) = 1, when two transaction A and B are exactly identical. sim(A, B) = 0, when two transactions A and B have no items in common.

The problem with this similarity measure is that it only considers the constitution of the transaction and not the temporal order of the pages accessed within the transactions. To solve this issue, a similarity measure called Sequence and Set Similarity Measure (S3M) is used. S3M is defined as below. [9] [11] [16]

$$S^3M(A,B) = p * \frac{LLCS(A,B)}{\max(|A|,|B|)} + q * \frac{|A \cap B|}{|A \cup B|}$$

Where,

LLCS(A, B) - Length of the Longest Common Subsequence of A & B

Max(|A|,|B|) - Maximum length between A & B

|A ∩ B| - Length of the intersection set of A & B

|A ∪ B| - Length of union set of A & B

p & q - Relative weights assigned to sequence and set similarities, p + q = 1

C. Similarity Upper Approximations

Let A be a transaction in the universe U. The similarity upper approximation of A is defined as follows.

$$\overline{R}(A) = \bigcup_{t \in A} R(t)$$

Where R (t) is the tolerance class of t.

The Similarity upper approximation is the set that contains the objects that are most similar to A. Which, in this context, means that a user who visits the pages in A, visits the pages present in $\bar{R}(A)$.

If now we find the similarity upper approximation of the set $\bar{R}(A)$ then we get the set $\overline{R\bar{R}}(A)$, which is called second similarity upper approximation and represents the objects that are most similar to $\bar{R}(A)$. We can get further similarity upper approximation of previous similarity upper approximation iteratively. At last we will be able to get the final similarity upper approximations. The stopping criterion here is the similarity of current and previous similarity upper approximations.

D. DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density based clustering algorithm. The algorithm grows regions with sufficiently high density into clusters and discovers clusters of arbitrary shape in spatial datasets with noise. It defines a cluster as a maximal set of density-connected points.

The complexity of DBSCAN is O (n^2), where n is the number of database objects.

The DBSCAN algorithm can identify clusters in large spatial data sets by looking at the local density of database elements, using only one input parameter. Furthermore, the user gets a suggestion on which parameter value that would be suitable. Therefore, minimal knowledge of the domain is required. However, one issue with DBSCAN is clusters that lie close to each other tend to belong to the same class. [8]

The reason why DBSCAN is an appropriate algorithm for identifying interesting user access patterns is based on following points.

- The logs in the access log files are very large in size.
- They contain the page access requests from very diverse group of users.
- The amount of noise present in the access log files is high.

DBSCAN is appropriate for the data having large size and noise. Unlike other clustering techniques, it is able to discover arbitrary shaped clusters so its accuracy is higher while working with dense and noisy data. [7][5][12][15]

E. K-NN classifier

K-Nearest Neighbours (K-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the K closest training examples in the feature space. K is a user-defined constant, and an unlabelled vector or test point is classified by assigning the label which is most frequent among the k training samples nearest to that query point. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its Neighbours, with the object being assigned to the class most common among its k nearest Neighbours (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest Neighbour. The distance matrices utilized with K-NN are Euclidean distance or cosine similarity. Usually the value of K parameter is taken as odd so as to avoid any ties while classification. [1][7]

The K-Nearest Neighbour (K-NN) algorithm is one of the simplest methods for solving classification problems. It has advantages like transparency, scalability, ease of implementation and reduced error rate. The complexity of K-NN algorithm is O(n).

F. Distance Measure

Categorical attribute is a non-numeric attribute such as colour or object. In the web usage data, the categories of the pages that user has access are recorded. To find distance among the transactions, distance measure of categorical attribute is used. It is defined as follows and based on the definition given in [20].

For given two transactions T_i and T_j having their access sequences as $\{x_1,\ldots x_n\}$ and $\{y_1\ldots y_m\}$

$$dist\,(T_i,T_j) = \sqrt{\sum_{k=1}^{max(m,n)} (x_{ik} - y_{ik})^2}$$

III. METHODOLOGY

A. Discovering the clusters of transactions

Let the transactions of msnbc.com that are to be used as training set be those in Table 3. Although the training set actually contains the clusters of these transactions, they are utilized in further computations.

Let the parameters of S^3M equation be p = 0.5 and q = 0.5

The S^3M of T_0 with that of T_6 is computed as

$$S^3M(T_0,T_6) = p * \frac{LLCS(T_0,T_6)}{Max(|T_0|,|T_6|)} + q * \frac{|T_0 \cap T_6|}{|T_0 \cup T_6|}$$

S^3M (T_0, T_6) = 0.5 * 2 / 2 + 0.5 * 2 / 2 = 1.0

And S^3M (T_0, T_{10}) = 0.5 * 2 / 7 + 0.5 * 1/ 2= 0.25 + 0.1428 = 0.3929

Likewise the S^3M matrix is

	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T0	0.0	0.0	0.0	0.0	**0.75**	0.0	**1.0**	0.0	0.0	0.0	**0.3929**	0.0	1.0
T1	0.0	**0.0**	**0.2222**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.2191**	0.0	0.0	0.0
T3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.1218**	0.0	0.0	0.0
T4	0.0	0.0	0.0	0.0	0.0	0.0	**0.75**	0.0	0.0	0.0	**0.3214**	0.0	**0.75**
T5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.2167**	**0.1218**	0.0	0.0	0.0
T6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.3929**	0.0	**1.0**	0.0	**1.0**
T7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.2167**	**0.1218**	0.0	0.0	0.0
T8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.101**	0.0	0.0	0.0
T9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**0.3929**
T11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 3 S3M Matrix

Now, applying the rough agglomerative clustering technique with relative similarity [11], the clusters are
$C_0 = \{T_0, T_4, T_6, T_{10}, T_{12}\}$
$C_1 = \{T_1,\}$, $C_2 = \{T_2\}$, $C_3 = \{T_3\}$,
$C_4 = \{T_0, T_4, T_6, T_{10}, T_{12}\}$, $C_5 = \{T_5, T_7\}$,
$C_6 = \{T_0, T_4, T_6, T_{10}, T_{12}\}$, $C_7 = \{T_5, T_7\}$,
$C_8 = \{T_8\}$, $C_9 = \{T_9\}$, $C_{10} = \{T_0, T_4, T_6, T_{10}, T_{12}\}$, $C_{11} = \{T_{11}\}$, $C_{12} = \{T_0, T_4, T_6, T_{10}, T_{12}\}$
And, applying DBSCAN with relative similarity (e = 0.2, MinPts = 3) [18], the clusters are

$C_0 = \{T_4, T_6, T_{12}\}$, $C_1 = \{T_1\}$, $C_2 = \{T_2\}$, $C_3 = \{T_3\}$, $C_4 = \{T_0\}$,
$C_5 = \{T_7\}$, $C_6 = \{T_6\}$, $C_7 = \{T_5\}$, $C_8 = \{T_8\}$, $C_9 = \{T_9\}$,
$C_{10} = \{T_{10}\}$, $C_{11} = \{T_{11}\}$, $C_{12} = \{T_{12}\}$

Considering the Levenshtein Distance (LD) as the inter cluster distance [11], the maximum LD of clusters in former case is 5 and that of later case is 3. Clearly, the later approach has better inter cluster similarity.

It can be concluded from the results of above two approaches that clusters formed using DBSCAN with rough set approach are crispier than that formed using rough agglomerative clustering.

B. Predicting the clusters of an unknown user

Let the unknown user transaction $T_{test} = \{6, 7, 7, 7, 6, 7\}$

Computing Euclid distance with categorical attribute,

$T_0 = \{1, 1\}$

Euclid $(T_{test}, T_0) = \sqrt{1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2} = 2.4494$

Here, T_{test}'s 1st page access category is on-air (6) and that of T_0 is FrontPage (1). As both are different, the difference is taken as 1. Same way other values are determined.

Euclid $(T_{test}, T_5) = \sqrt{1^2 + 1^2 + 1^2 + 1^2 + 1^2} = 2.2360$

Euclid $(T_{test}, T_2) = \sqrt{1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2} = 3$

Likewise, Euclid distance of T_{test} with all other transactions in sorted order are as shown in Table 1. Their line graph is shown in figure 3.

TABLE I
EUCLID DISTANCE OF T_{test} WITH TRAINING TRANSACTIONS

Transaction	Distance with T_{test}
T_5	2.2360
T_7	2.2360
T_8	2.2360
T_0	2.4494
T_1	2.4494
T_3	2.4494
T_4	2.4494
T_6	2.4494
T_{11}	2.4494
T_{12}	2.4494
T_{10}	2.6457
T_2	3.0
T_9	3.4641

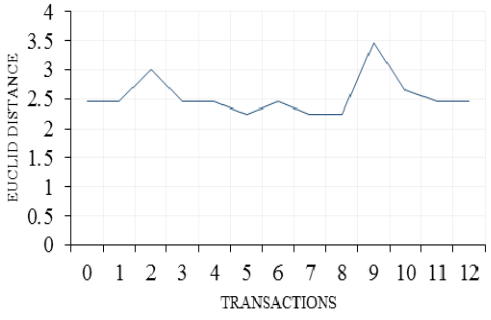

Figure 4 Euclid Distance of T_{test} with training transactions

1) Clusters of rough agglomerative clustering as the training set:

- K = 1
  Nearest neighbors NN = $\{T_5\}$
  $T_5 \in \{C_5, C_7\}$
  Thus, $T_{test} \in \{C_5, C_7\}$
  C_5 and C_7 are the predicted clusters of transaction T_{test}
- K = 3
  Nearest neighbors NN = $\{T_5, T_7, T_8\}$
  Majority group which belongs to same cluster, M = $\{T_5, T_7\}$
  $M \in \{C_5, C_7\}$
  $T_{test} \in \{C_5, C_7\}$

Here also, C_5 and C_7 are the predicted clusters of transaction T_{test}

2) Clusters of DBSCAN with rough set approach as the training set:

- K = 1
  Nearest neighbors NN = $\{T_5\}$
  As $T_5 \in C_7$
  $T_{test} \in \{C_7\}$

The predicted cluster of T_{test} is C_7

- K = 3
  Nearest neighbors NN = $\{T_5, T_7, T_8\}$
  Not able to find majority class M as each of transaction in NN belongs to different cluster.

Solutions:

- Use lesser value of K. Ex. K=1
- Default cluster as predicted cluster. The default cluster is the cluster which contains maximum numbers of transactions. C_0 is the default cluster in this example.

IV. ALGORITHMS

A. DBSCAN with rough set approach

Input
T: A set of n transactions $\in$ U
d: Threshold $\in$ [0, 1]
e: eps-neighborhood $\in$ [0, 1]
MinPts: Minimum number of Neighborhood points
Output
A set of clusters C
Method
Step 1. M = computeSimilarityMatrix (T, p)
Step 2. C = DBSCAN (T, M, e, MinPts)
Step 3. For each clusters $\in$ C
find the next similarity approximations S.
Step 4. for each transactions i and j
if Si != Sj go to Step 3
Step 5. Return C

B. K-NN Classifier

Input
C: Initial Clusters
T: Initial Transactions
Ttest: Unknown user transaction
K: Value of K parameter

Output

$C_{predicted}$: Set of Predicted Clusters

Method

Step 1. Compute Categorical Euclidian Distance of T_{test} with the set T

Step 2. Find K – Nearest Neighbours

Step 3. Determine the Majority Class from the members of K – Nearest Neighbours

Step 4. If (Majority Class exist)

a. Find the clusters containing the Majority Class

b. If (no such cluster is found)

Put default cluster in $C_{predicted}$

Else

Put the clusters found in $C_{predicted}$

Else

Put the default cluster in $C_{predicted}$

Step 5. Return $C_{predicted}$

V. EXPERIMENT

A. Description of Dataset

We collected the dataset from the UCI data repository that consists of several logs from msnbc.com for the month of September 1998. Each sequence in the dataset corresponds to the page views of a user during that 24 hours period. Each event in the sequence corresponds to a user request for a page.

There are 17 page categories: FrontPage, news, tech, local, opinion, on-air, misc, weather, health, living, business, sports, summary, bulletin board service, travel, msn-news, and msnsports.

Each category is associated in order with an integer starting with 1. For example, FrontPage is associated with 1, news with 2, and tech with 3. First 13 data sequences are shown in the table II.

TABLE II
PAGE ACCESS TRANSACTIONS

Transaction	Order of user page visits
T_0	1 1
T_1	2
T_2	3 2 2 4 2 2 2 3 3
T_3	5
T_4	1
T_5	6
T_6	1 1
T_7	6
T_8	6 7 7 7 6 6 8 8 8 8
T_9	6 9 4 4 4 10 3 10 5 10 4 4 4
T_{10}	1 1 1 11 1 1 1
T_{11}	12 12
T_{12}	1 1

B. Environment

The tool used to develop the algorithms is NetBeans 8.0.2. The algorithms are developed using java programming language with JDK 7. The implementation was done on Pentium i3 processor clocked at 1.90 GHz with 4 GB RAM and operating with 64 bit Windows 10 Operating System.

C. Results

The execution of the clustering algorithm on transactions T_0 to T_{12} with parameters eps=0.2 and minPts=3 leads to the clusters same as shown in the section III. Now using these clusters for prediction of clusters for transactions T_{14} to T_{19} using K-NN with k=1 leads following results as shown in Table III. Here, the highlighted clusters in the last column represents the difference in the result by DBSCAN with rough set approach against rough agglomerative clustering.

TABLE II
PREDICTED CLUSTERS OF T_{test} WITH ROUGH AGGLOMERATIVE CLUSTERING AND DBSCAN WITH ROUGH SET APPROACH AS TRAINING SETS

Transaction	Predicted clusters (Rough agglomerative clustering)	Predicted clusters (DBSCAN with rough set approach)
T_{14}	C_5 ,C_7	**C_7**
T_{15}	C_1	C_1
T_{16}	C_6	**C_0**
T_{17}	C_1	C_1
T_{18}	C_1	C_1
T_{19}	C_1	C_1

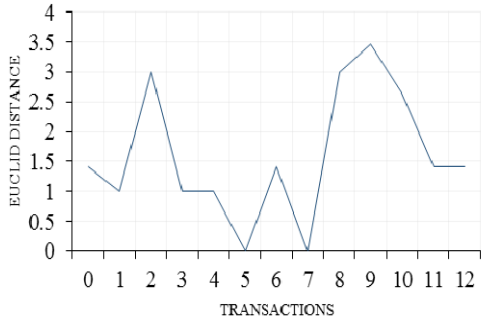

Figure 5 Euclid distance of T_{14} with Training Transactions

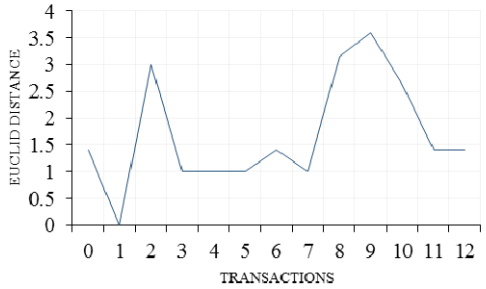

Figure 6 Euclid distance of T_{15} with Training Transactions

Figure 5 and Figure 6 represents the Euclid distance for categorical attributes of transactions T_{14} and T_{15} with the Training transactions T_0 to T_{12}. These distances are used to compute the predicted clusters of Table III by finding k-nearest neighbours.

VI. Analysis of Results and Summary of Findings

We have executed the algorithms on data sequences of msnbc.com transactional dataset of the month September 1998 having different number of transactions. The reasons for choosing the dataset is that it is a standardized one and it is best suited with the nature of this research. For each run of the algorithm, the number of clusters formed and total execution time taken by the clustering algorithm are recorded. The execution time for each run is shown in table IV.

TABLE III
EXECUTION TIME (IN SECONDS) FOR EACH RUN OF ALGORITHM

Transactions	Rough agglomerative	DBSCAN with RelSim
100	23	17
200	635	372
300	2967	1924
400	8037	7170
500	19757	15991

Figure 7 shows the graphical representation of the same. It shows the comparison of execution time of DBSCAN with rough set approach with that of Rough agglomerative clustering for same number of transactions. From these results it can be concluded that for same number of transactions and same execution environment the DBSCAN with rough set approach performs considerably better than Rough agglomerative clustering in terms of execution time.

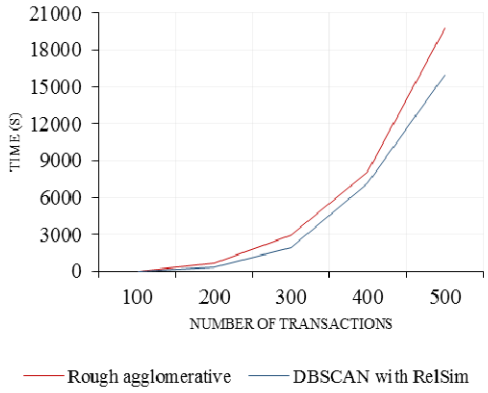

Figure 7 Execution time (in seconds) against Number of Transactions

VII. Conclusions

Rough set theory is a mathematical approach to handle inconsistencies and ambiguity present in the data. It represents an ambiguous set into two sets called lower approximation and upper approximation. Web usage data contains access information of users of the web server in the form of web access logs. The usage data is ambiguous and noisy. Utilizing rough sets with a clustering algorithm which can handle noise well we can create a solution that divides the user transactions hence, the users in to hard and unambiguous groups. DBSCAN with rough set approach as compared to rough agglomerative clustering creates hard clusters. These clusters are improved than those found using rough agglomerative clustering. The performance of former against later is also considerably better. As a part of the application of this approach, the clusters are applied as the training set to a predictor based on K-Nearest Neighbor classifier. The experimental results suggests that DBSCAN with rough set approach works better here, too.

As the future work, this solution can be utilized to create a recommendation system or a web personalization system.

References

[1] K. Sudheer Reddy, G. Partha Saradhi Varma, and M. Kantha Reddy, "An Effective Pre-processing Method for Web Usage Mining", *International Journal of Computer Theory and Engineering*, Vol. 6, No. 5, 2014.

[2] Sujith Jayaprakash , Balamurugan, "A Comprehensive Survey on Data Pre-processing Methods in Web Usage Mining" , *International Journal of Computer Science and Information Technologies*, Vol. 6, 3, 2015.

[3] Sanjeev Dhawan, Swati Goel, "Web Usage Mining: Finding Usage Patterns from Web Logs", American International Journal of Research in Science, Technology, Engineering & Mathematics.

[4] Nirali H.Panchal, Ompriya Kale, "A Survey on Web Usage Mining", *International Journal of Computer Trends and Technology (IJCTT),* vol. 17, no. 04, Nov. 2014.

[5] Karuna Katariya, Rajanikanth Aluvalu, "Agglomerative Clustering in Web Usage Mining: A Survey", *International Journal of Computer Applications,* vol. 89, no. 08, 2014.

[6] D. Jayalatchumy, Dr. P.Thambidurai, "Web Mining Research Issues and Future Directions – A Survey", *IOSR Journal of Computer Engineering (IOSR-JCE),* vol. 14, 03, 2013, 20-27.

[7] Jiawei Han, Michaline Kamber, Jian Pei, *Data mining concepts and techniques*, 3rd ed., Elsevier Inc., 2012.

[8] Shaily G.Langhnoja, Mehul P. Barot, Darshak B. Mehta, "Web Usage Mining Using Association Rule Mining on Clustered Data for Pattern Discovery", *International Journal of Data Mining Techniques and Applications*, vol. 02, 01, 2013.

[9] Supriya Kumar De, P. Radha Krishnab, "Clustering web transactions using rough approximation", *Fuzzy Sets and Systems, ELSEVIER*, 2004, 131-138.

[10] G.Ramani**, "**Rough set with Effective Clustering Method", International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, 2, 2013.

[11] Pradeep Kumar, Radha Krishna, Supriya Kumar de, Raju Surampudi Bapi, "Web usage mining using rough agglomerative clustering", Researchgate, 2005.

[12] Harish Kumar, Anil Kumar, "Clustering algorithms employed in web usage mining: An overview", *Proceeding of the 5th National Conference, INDIACom, Computing for national development*, 2011.

[13] P. Nithya, Dr. P. Sumathi, "A survey on Web Usage Mining: Theory and Applications", *International Journal of Computer Technology and Applications,* vol. 3, 4, 2012.

[14] Mohammed Hamed Ahmed Elhiber, Ajith Abraham, "Access Patterns in Web Log Data: A Review", *Journal of Network and Innovative Computing*, vol 1, 2013, pp. 348-355.

[15] Deepti Sisodia, Lokesh Singh, Sheetal Sisodia, Khushboo Saxena, "Clustering Techniques: A Brief Survey of Different Clustering Algorithms", *International Journal of Latest Trends in Engineering and Technology*, vol. 1, 3, 2012, pp. 82-87.

[16] Ms K.Santhisree, Dr A.Damodaram, "Clustering on Web usage data using Approximations and set similarities", *International Journal of Computer Applications*, vol. 1, 4, 2010.

[17] Rajhans Mishra , Pradeep Kumar , "Clustering Web Logs Using Similarity Upper Approximations", *International Journal of Machine Learning and Computing*, vol 2, 3, june 2012.

[18] K. Santhisree, A. Damodaram, SV Appaji. "An Enhanced DBSCAN Algorithm to Cluster Web usage Data using Rough Sets and Upper Approximations", *International Journal of Computer Science & Communication,* vol. 1, 1, January – June 2010, pp. 263 – 265

[19] Pawan Lingras, Georg Peters, "Applying Rough Set Concepts to Clustering", *Rough Sets: Selected Methods and Applications in Management and Engineering*, Advanced Information and Knowledge Processing, Springer, 2012.

[20] D.A. Adeniyi, Z. Wei, Y. Yongquan, "Automated web usage data mining and recommendation system using K-Nearest Neighbour (KNN) classification method", *Applied Computing and Informatics*, ScienceDirect, Vol. 12, 2016, pp. 90-108.