Property Preservation in Reduction of Data Volume for Mining: A Neighborhood System Approach

Ray R. Hashemi
Department of Computer Science
Armstrong State University
Savannah, GA, USA
Rayhashemi@gmail.com

Azita Bahrami
IT Consultation
Savannah, GA, USA
Azita.G.Bahrami@gmail.com

Nicholas R. Tyler
Department of Biology
Armstrong State University
Savannah, GA, USA
Romtinian@gmail.com

Matthew Antonelli and Bryan Dahlqvist
Department of Computer Science
Armstrong State University
Savannah, GA, USA
Mattr.Antonelli@gmail.com
n_bryan28@hotmail.com

***Abstract*— The sheer volume of the very large datasets is the major obstacle in mining of the data because the size of the dataset is above the handling abilities of the traditional methodologies. A considerable vertical reduction over and beyond the reduction prescribed by pre-mining processes is needed to overcome the problem. However, the reduced version of the dataset ought to preserve the intrinsic properties of the original dataset in reference to a specific mining goal (a robust reduction); otherwise, it is a useless reduction. This research effort introduces and investigates the neighborhood system as a robust data volume reduction methodology in reference to the mining goal of "prediction". Two well-known prediction algorithms of ID3 and Rough Sets are employed to determine the perseveration of intrinsic properties in the reduced datasets. The results obtained from 10 pairs of training and test sets revealed that the proposed reduction methodology is a robust one and it also reduces noise in data which in turn improves the prediction outcomes. The average percentage measures of: (i) the correct prediction increases by 26%, (ii) the false positive decreases by 36%, (iii) the false negative decreases by 89%, and (iv) the unpredictable objects increases by 136% which is the indicative of a reliable system. Prediction of no decision for an object is always preferred over prediction of a false positive or a false negative decision. The neighborhood-based reduction system also increases the granularity of the dataset which is different from the increase in the granularity through the use of a generalization process.**

Keywords—Data Mining; Big Data; Data Volume Reduction; Neighborhood System; Property Preservation; Organic Discretization.

I. INTRODUCTION

A very large dataset may be mined for the purpose of association analysis, concept analysis, decision support analysis, market analysis, and prediction, to name a few. The sheer volume of a very large dataset is the major obstacle in mining the data because the size of the dataset is beyond handling abilities of the traditional methodologies. Any methodology used for reducing the size of the dataset must be able to preserve the *intrinsic properties* of the very large dataset; otherwise, the methodology is not a robust one.

To remove, or at least ease, the volume obstacle, partitioning methodologies have been contemplated [1]. In any partition-based methodology, the very large dataset is divided into partitions either randomly or based on some criteria suggested by the mining goal. The mining of each partition takes place separately. However, the mining outcome (intrinsic properties) of a very large dataset is not equivalent of the union of the intrinsic properties of the individual partitions. Reader needs to know that the parallel processing plays a big role in mining of very large datasets and datasets are segmented for use by the parallel processor [2]. This segmentation is different from partitioning because during the segmentation process the dataset is perceived as one entity, whereas the partitioning process perceives each partition as a separate dataset.

Clustering-based methodologies may also reduce the volume of data [3]. The common practice is that a cluster of records of the very large dataset is replaced by the seed of the cluster. The inclusion of a record in a cluster is based on the fact that the sum of its attribute distances from the corresponding attributes of the seed is less than a threshold distance. The problem with clustering is that it is influenced by the sum of the individual attribute's differences and not by the differences of the individual attributes. As a result, a cluster satisfies a condition that does not guaranty the true homogeneity of its record members. Replacing a cluster of non-homogenous records with its seed has a dire effect on the preservation of the properties of the large dataset.

In this paper, we propose a methodology, *neighborhood system*, for volume reduction of very large datasets. We also empirically show that the reduction methodology is a

 ISBN: 978-1-61208-358-2

robust one. That is, the reduced dataset preserves the intrinsic properties of the original dataset.

The organization for the rest of the paper is as follow: The previous work is presented in Section two. The methodology is the subject of Section three. The empirical results are covered in Section four. The conclusion and future research are discussed in Section five.

II. PREVIOUS WORKS

The data reduction has been explored by researcher for four different purposes of *data storage*, *data transmission*, *data presentation* and *data mining*. For data storage, basically, data is compressed to take less space. A compressed dataset may be decompressed as needed. Numerous data compression techniques have been reported in literature [4][5]. Some compression techniques are lossy and some are lossless. Use of lossy techniques compresses data in a way that it cannot turn completely into its original form upon applying the decompression process. In contrast, data compressed by using the lossless compression techniques turned into the original dataset after decompression.

For data transmission, data is reduced during its preparation for the transmission and usually returns to its original form at the destination. For example, prior to transmission of an image through a communication channel, the image is reduced to lower the communication time and be adapted to the communication channel limitations [6][7].

For data presentation, data is reduced using different way of its presentation. For example, visualization of data presents data in a reduced form [8][9]. As another example, collection of a high volume of raw data is used for building a product. By doing so, the final product becomes the reduced version of the raw data [10][11].

For data mining, data is reduced horizontally and vertically prior to applying any data mining methodology. The horizontal data reduction means removing the redundant attributes from a dataset. Entropy analysis, correlation analysis, relevancy analysis, and rough sets are some of the well-known methods for performing the horizontal reduction [12][13][14][15]. The vertical reduction reduces the number of records in a dataset. This is done through collapsing the duplicated records and in some cases removal of conflicting records. Such reduction is a part of the pre-mining process and the reduced datasets often have slightly less number of records than the original datasets. For very large datasets, a considerable vertical reduction in addition to the vertical reduction prescribed by the pre-mining process is needed. However, the reduced version of the dataset ought to preserve the intrinsic properties of the original dataset; otherwise, it is a useless reduction.

In this paper, we propose and investigate a robust vertical reduction methodology that is able to (a) reduce the size of dataset beyond pre-mining reduction and (b) preserve the intrinsic properties of the original dataset. To the best of our knowledge, there is no such investigation reported in the literature.

III. METHODOLOGY

We present, first, the neighborhood system as a new methodology for reducing the volume of a dataset. Second, we introduce formal definition of the *intrinsic properties* (or simply *properties*) of a dataset along with the methodology for testing the property preservation. Finally, we present the organic discretization in support of property preservation.

A. The Neighborhood System

A dataset is a collection of records and each record has n attributes $U = \{A_1, \ldots, A_n\}$. Consider records R_i: $(v_1, \ldots, v_n)$ and R_j: $(v'_1, \ldots, v'_n)$, (values of v_k and v'_k belong to attribute A_k). R_j is the *neighbor* of R_i in reference to U, if $|v_i - v'_i| \leq r$ (for $1 \leq i \leq n$). r is a radius threshold. All the neighbors of R_i within a given dataset make the *neighborhood* of R_i. It is true to say that every record is also a member of its own neighborhood. The following notation is used to denote the neighborhood of R_i: $N(R_i)_{[U, r]}$. If R_j is in $N(R_i)_{[u,r]}$, then R_i is also in $N(R_j)_{[u,r]}$.

Since the threshold radius can take many different values, the record R_i may have many, not necessarily distinct, neighborhoods. This is true for all the records in the dataset. The neighborhoods of every record of a dataset are collectively referred to as a *neighborhood system* of the dataset.

Hashemi et al. [16] divide the neighborhood system for each record into three regions of *closest*, *closer*, and *close* neighborhoods. These regions for R_i are defined as Closest $(R_i) = N(R_i)_{[U, r=0]}$, Closer $(R_i) = N(R_i)_{[U, r=a]}$, and Close $(R_i) = N(R_i)_{[U, r=b, \text{ where } b>a]}$. The three regions are also known as *the workable neighborhoods* of R_i.

TABLE I: A DATASET.

Records	A1	A2	A3	A4	A5
R_1	1	2	1	3	4
R_2	1	1	2	2	2
R_3	2	2	3	1	2
R_4	1	2	1	3	4
R_5	2	3	2	2	3
R_6	3	1	3	1	2
R_7	2	1	1	2	3
R_8	3	2	2	3	3

As an example, consider the dataset of Table 1. For the record R1, the workable neighborhoods are:

$\text{Closest}(R_1) = \{R_1, R_4\}$

$\text{Closer}(R_1) = \{R_1, R_4, R_5, R_7\}$

$\text{Close}(R_1) = \{R_1, R_2, R_4, R_3, R_7, R_5, R_6, R_8\}$

For identifying the closer and close neighborhoods of R_1, we use a = 1 and b = 2. Therefore, the closest, closer, and close neighborhoods of R_1 include those records of the dataset that their attribute values differ from their

corresponding attribute values in R_1 by zero, absolute value of one, and absolute value of 2, respectively.

Since, in this research effort, the goal of the mining of a very large dataset is to perform "prediction", we assume that each record has an extra attribute of *decision*, Table 2.

The decision attribute does not play any role in finding a neighborhood. However, the decision attribute is used to assign a *certainty factor* to any neighborhood of interest. For example, the certainty factor for the Closer(R_i) is α = N/|Closer(R_i)|, where N is the number of records in the Closer(R_i) who have the same decision value as R_i. As a result, the certainty factor for the Closest(R_1), Closer(R_1), Close(R_1) are 1, 1, and 5/8, respectively.

TABLE II: A DATASET WITH A DECISION ATTRIBUTE.

Records	A1	A2	A3	A4	A5	Decision
R_1	1	2	1	3	4	1
R_2	1	1	2	2	2	0
R_3	2	2	3	1	2	0
R_4	1	2	1	3	4	1
R_5	2	3	2	2	3	1
R_6	3	1	3	1	2	1
R_7	2	1	1	2	3	1
R_8	3	2	2	3	3	0

1) Record Tree

Let us focus on the record R_i and its closer neighborhood. Initially, the *record tree* of the R_i is the presentation of Closer(R_i) in form of a tree for which R_i is the root and the neighbors of R_i are the children. The tree is assigned a *certainty factor* and a *signature*. The certainty factor of the tree is the same as the certainty factor of the Closer(R_i). The signature of the tree is a record with the same number of attributes as R_i and the value for attribute A_m of the signature is the average of values of attribute A_m for all the records in the record tree.

Each child, C_i, of the tree is expanded as a new sub-tree by the records in Closer(C_i). The new sub-tree is pruned based on the following criteria:

a. If a child of C_i is already appeared as a node somewhere in the tree, the child is pruned.
b. If the Euclidean distance of a child of C_i from the signature of the tree is greater than a given threshold value, the child is also pruned.

After the expansion of Ci, if any of its children survived the pruning process, the record tree of R_i becomes a new tree with a new certainty factor and a new signature. The process of expansion of the new record tree for R_i continues in a breadth-first fashion until it cannot be expanded any longer. All the records that are part of the totally expanded record tree of R_i will not have their own record trees and cannot be a part of another record tree. However, the building of the record trees for the remaining records of the dataset is a continual process.

Selection of R_i for building its record tree is not a random act. R_i is selected such that its closer neighborhood has the highest certainty factor among all the closer neighborhoods of the dataset. In case of a tie, R_i has the highest cardinality. If having a tie persists, R_i is selected randomly among the qualified records.

The signature of the record tree for Closer(R_i) acts as a representative of all the records in the neighborhood and replaces all of them. Let us assume that the process of building record tree for the records of a dataset produces T record trees. The ratio of |dataset|/T is the *reduction factor*. For the dataset in Table 2 and for the Euclidean distance threshold of 3.7, only two record trees are produced: see Figure 1. Therefore, the reduction factor is 8/2 = 4.

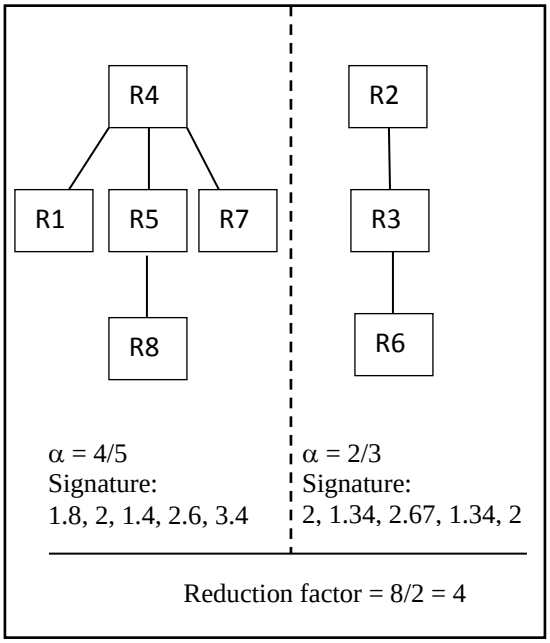

Figure 1. Record trees for the dataset of Table 2 along with their certainty factors and signatures.

B. Properties of a Dataset

So far, the proposed neighborhood system is able to reduce a given very large dataset, V, by factor of K into a new dataset, V'. In this section, we define the intrinsic properties and describe the methodology for checking the preservation of the properties of V by V'.

The properties of a very large data set are a sextuple, (V, G, A, F, Q, E), where:

V is a very large dataset,
G is the goal of the mining process
A is a methodology used for reaching the goal of G.
F is resulting set of findings applying A on V,
Q is the quality measure for F. The quality of F is usually measured by using another dataset (E).
E is the entity involved in measurement of the quality for F.

The quality measure of F needs an explanation. Let us assume G is "prediction" and A is the "ID3" algorithm [17]. The outcome of applying A on V is a set of prediction rules, F. The quality measure for F is the quality of the prediction for the test records using F. Thus, the test set is E.

As another example, let us assume G is "basket analysis" and A is the "Apriori" algorithm [18]. The outcome of applying A on V is a set of frequent itemsets, F. The quality measure of F is the collection of the quality measures for each frequent itemset which is presented in pair of (*support, confidence*). Calculation of support and confidence are done using records in V. Therefore, V is the entity involved in quality measure of F.

Definition: Let V' be a reduced version of V and the mining goal for both V and V' be the same. Let also A be a well-established algorithm for reaching the goal. In addition, let F and F' be the two sets of findings produced by applying A on V and V'. In addition, let Q and Q' be the quality measures of F and F', respectively, calculated using the same entity E. The properties of V and V' are: *(V, G, A, F, Q, E)* and *(V', G, A, F', Q', E)*. If Q' = Q, then the properties of V has been preserved by V'. If Q' > Q then V' has not only preserved the properties of V but also reduced noise in data.

Let us assume that the algorithm A cannot be applied on V' due to the fact that data in V' is continuous, whereas data in V is discretized (required by the algorithm A). To remove this obstacle, either data in V' needs to be discretized or algorithm A needs to be replaced by another algorithm that can process both discrete and continuous data. The first option is more logical because it does not limit the list of algorithms that can be applied on V'. As a result, we introduce our own discretization methodology named *organic discretization* in the following sub-section.

C. Organic Discretization

The majority of the discretization methodologies, reported in literature, have an artificial discretization theme [19][20]. For example, the interval between the maximum and minimum values of an attribute is divided into a number of equal width smaller intervals and each small interval is assigned a discrete value that replaces all the values within the small interval. Such discretization is artificial and does not consider any characteristics of the values of the attribute. Although some of the methodologies such as bin-based and radius-based try to ease the problem, but they cannot avoid artificially discretizing the data [2]. There are more sophisticated discretization methodologies that are so labor intensive that their use is not cost effective [2].

In this section, we propose an organic discretization methodology that uses the closeness of values in an attribute for discretization. The methodology is simple and organic. Because of that the intervals represented by discrete values do not have necessarily the same width and the width of each interval is decided by the data itself.

In this methodology, the values of the attribute are sorted in ascending order and the differences between every two adjacent values are measured. A user selects a preferred small difference, P_d. The end point of a current interval is decided based on the differences between the value located in locations L and L+1 in the list of sorted values and P_d. If P_d is zero, then every unique value belongs to a new interval. If P_d is too large, then the entire attribute becomes one interval. The best choice for P_d to discretize an attribute of dataset V' is to generate the same or close number of intervals—and therefore, discrete values—for the attribute as there is for the corresponding attribute in V. The reason stems from the fact that some mining algorithms choose attributes with more discrete values over those with less number of attributes or vice versa. Since the goal is to investigate the preservation of the properties in reduced datasets, we want to remove any biases causing by the number of discrete values.

The following algorithm provides the details of the organic discretization approach:

Algorithm Organic

Input: A dataset with n attributes of A1, . . ., An. Data of the dataset is continuous. Two threshold values of t_d and t_{count}.

Output: The discretized dataset.

Step1. Repeat for each attribute A_i

Step2. B = Ai, sorted in ascending order.

Step3. C[i] = ABS(B[i]-B[i+1]).

Step4. Locate in C those elements with value > t_d and Collect their indices in array D.

Step5. top = 1; bottom = 1; Count = 0; //Top and bottom are pointers pointing to the first element of interest in array B and first element of interest in D;

Step6. Repeat Steps 7 to 9 while top < |B|;

Step7. If D= ∅, then bottom = |B|;

Step8. Those values in array B from B[top] to B[D[bottom]] make an interval, Int, represented by a discrete value which is the median of the values in the interval; count++;

Step9. top = top + |inter|; Remove the first element of D;

Step10. If count > t_{count}, then increase t_{count}; go to Step2;

Step11. End;

The variation of the algorithm may be considered by the choosing a different value to represent the interval produced in Step 8. We used the median value to represent the interval.

IV. EMPIRICAL RESULTS

In a glance, we: (i) generate 10 pairs of the training and test sets out of the original dataset, (ii) generate the reduced version of the same 10 training sets using the neighborhood

system and the organic discretization approaches, (iii) select a mining goal and a well-known algorithm to achieve it. If the average results produced by applying the well-known algorithm on the second ten pairs (the reduced ones) is the same or better than the average results produced by applying the algorithm on the first ten pairs (the original ones), then the reduced version of the training sets has preserved the intrinsic properties of the original dataset; Otherwise, the intrinsic properties have been damaged and the reduction methodology is not robust.

To provide the details of the process, the "prediction" is our mining goal and the algorithm to achieve the goal is the well-known ID3 algorithm. We have a dataset with 1000 records and each record has 8 attributes. Each one of the first seven attributes has six possible discrete values. The last attribute is a decision and it has two possible values of 1 and 0. One may point out that the dataset is not a very large dataset. However, here our goal is to show the proof of concept.

Ten percent of the records with decision 1 and ten percent of the records with decision zero have been set aside to make one test set. Among the remaining m records, m1 of them has decision one and m2 of them has decision zero (m = m1 + m2 and m2 <m1). We pick m2 records out of the m1 records with decision one along with the entire m2 records with decision zero and make the training set.

By repeating the same process, we generated 10 pairs of the training (Tr) and test (Ts) sets such that $Ts_i \cap Ts_j = \emptyset$ (for i = 1 to 10, j = 1 to 10, and i ≠ j) and $Tr_i \cap Ts_i = \emptyset$ (for i = 1 to 10). Since the original dataset is made up of the discrete values, so the training and test set pairs.

We have also generated:

1. A reduced version of each training set by applying the neighborhood methodology on the set (the reduction factor of the training sets was varying from 3.2 to 4.55 for different training sets). As a result, we produced 10 reduced training sets. The data of the reduced training sets were no longer discrete values.
2. A discretized version of each reduced training set by applying the organic discretization methodology on the set. It was clear that the new discretized values in the training set did not have the same meaning as the discrete values in the corresponding test set. Therefore, the discretization intervals of data established by the organic discretization of the training set were used to discretize the original test set.

To sum-up, we ended up having 10 pairs of the training and test sets build out of the original dataset and 10 pairs of the same training and test sets with the new discrete values influenced by the neighborhood methodology. The first and the second 10 pairs are referred to as the *Original* and *Reduced* sets, respectively.

To investigate the preservation of the properties, we took the following step for each training and test pair in the Original and Reduced sets:

- ID3 was applied on the training set and the prediction rules were obtained and used to predict the decision for the records of the test set. The quality of the prediction was measured by calculating the percentage of the number of correct predictions, false positives, false negatives, and not predictable records. The quality of the prediction for the Original pairs and Reduced pairs are shown in Table 3 and Table 4, respectively.

TABLE III: THE QUALITY MEASURES OF THE PREDICTION PROCESS FOR THE ORIGINAL SET USING ID3.

Original Pairs	% Correct Predictions	% False (+)	% False (-)	% Not-predictable
1	47.6	48	5	0
2	59.5	29	11	0
3	64.3	34	2	0
4	57.2	31	11	0
5	40.5	52	7	0
6	61.9	24	11	2
7	78.6	12	9	0
8	47.6	41	11	0
9	54.8	26	11	7
10	52.4	36	9	2
Avg.	56.4	33.3	9.7	1.1

TABLE IV: THE QUALITY MEASURES OF THE PREDICTION PROCESS FOR THE REDUCED SET USING ID3.

Reduced Pairs	% Correct prediction	% False (+)	% False (-)	% Not-predictable
1	66.7	24	0	10
2	76.2	19	0	5
3	66.7	26	1	7
4	71.4	17	1	11
5	71.4	21	1	7
6	76.2	12	2	10
7	69	14	0	17
8	66.7	24	0	10
9	64.3	12	1	23
10	73.8	12	2	12
Avg.	70.24	18.1	0.8	11.2

Since the average performance of ID3 on the Reduced set is much better than the average performance of ID3 on the Original set, the intrinsic properties of each test set has been preserved.

One may raise the following question: Is the property preservation possible using a prediction algorithm other than ID3? To answer this question we also conducted the same experiment using the Rough Sets algorithm [15][21][22][23]. The results for the Original and Reduced sets are shown in Table 5 and Table 6, respectively.

 ISBN: 978-1-61208-358-2

The Average prediction performance of the Rough Sets approach on the Original and the Reduced sets support the findings through the use of ID3.

V. CONCLUSION AND FUTURE RESEARCH

The results of Tables 3, 4, 5, and 6 reveal that the proposed reduction methodology preserves the intrinsic properties of the original dataset. Considering all four tables, the average percentage measure of: (i) the correct prediction increases by 26%, (ii) the false positive decreases by 36%, (iii) the false negative decreases by 89%, and (iv) the unpredictable records increases by 136% which it is the indicative of a reliable system. Prediction of a "no decision" for an object is always preferred over prediction of a false positive or a false negative decision.

TABLE V: THE QUALITY MEASURES OF THE PREDICTION PROCESS USING ORIGINAL SET AND ROUGH SETS.

Original Pairs	% Correct Predictions	% False (+)	% False (-)	% Not-predictable
1	40.2	38	11	10
2	53.1	22	16	9
3	60.8	14	15	10
4	41.7	27	11	10
5	40.4	38	9	12
6	55.7	24	12	8
7	62.9	12	15	10
8	75.8	12	7	5
9	66.8	16	10	7
10	40.1	36	19	5
Avg.	53.75	23.9	12.5	8.6

TABLE VI: THE QUALITY MEASURES OF THE PREDICTION PROCESS USING REDUCED SET AND ROUGH SETS.

Reduced Pairs	% Correct prediction	% False (+)	% False (-)	% Not-predictable
1	68.3	18	5	9
2	72.9	19	2	7
3	62.2	28	1	9
4	70.8	17	2	10
5	68.2	24	2	5
6	75.1	13	2	10
7	69	14	1	16
8	63.1	26	1	9
9	64.3	13	0	23
10	67.8	14	0	19
Avg.	68.17	18.6	1.6	11.7

To explain an interesting observation, let us briefly talk about noisy data which is a synonym for the erroneous data [1]. Error (noise) in data is resulting from corruption of data at the time of collection and or inputting. The noise in data is considered as an obstruction in any data mining process including prediction. The improvement of the prediction results for the Reduced set, by both ID3 and Rough Sets algorithms, indicates the fact that the data in the Reduced set has less noise than data in the Original set. Therefore, the proposed data reduction methodology not only preserves the intrinsic properties of the Original set but it also decreases the noise in the set.

The neighborhood-based reduction system also increases the granularity of the dataset which is different from the increase in the granularity through the use of a generalization process. To explain it further, let us assume that a dataset contains the monthly profit reported for a given company for duration of one year. This dataset has 12 records (one per month). One may add up the monthly profits for each quarter to express the quarterly profit. In this case, the dataset is reduced and it has only four records. The number of records may change into only 2 records, if bi-annual profits is sought. The reduction of the records provides different granules in each case. In the first and the second reductions each granule represents quarter profits and bi-annual profits, respectively. The reductions are also known as the representations of the profits for two *foot-steps* ("quarter" and "bi-annual") within the *concept hierarchy* of the time.

The prediction rules obtained from the higher granules may not preserve the properties of the dataset. The reason stems from the fact that (i) a higher granule ignores the details of lower granules and (ii) the foot-steps in a concept hierarchy are natural steps within the domain of the interest (in our example time domain) and does not have anything to do with the closeness of values of the records' attributes within the foot-step.

In contrast, the granularity provided by the proposed reduction methodology is only based on the closeness of values of the records' attributes. The foot-step based granularity still can be applied to the granules delivered by the proposed reduction system.

One of the challenges in this research effort was the selection of a representative for the interval produced in Step 8 of the Algorithm Organic. On the whole, there are seven possible options; thus, seven variations of the algorithm may be used. The options are: (i) the median value within the interval when the number of records, n, in the interval is odd, (ii) the $(n/2)^{th}$ value when n is even, (iii) the $[(n/2)+1]^{th}$ value when the n is even, (iv) average of $(n/2)^{th}$ value and $[(n/2)+1]^{th}$ value when n is even, (v) the first number in the interval, (vi) the last number in the interval, and (vii) average of all the values in the interval. We have chosen options (i) and option (ii) for the cases that n is odd and even, respectively. The methodology used for selecting these two options was the "trial-and-error" approach.

It was also noted that the robust reduction methodology for mining the prediction rules, may not be able to preserve the properties of the dataset for another mining goal –say, association analysis. For example, let us assume that we are interested in learning about the correlation between two values of "a" and "b" that belong to two different attributes

 ISBN: 978-1-61208-358-2

of a given dataset. The reduction process may change the value of "a" into possibly m new and different values. The value "b" may also be changed into possibly k new and different values (m and k are not necessarily equal). As a result, finding the correlation between values of "a" and "b" within the reduced dataset may be a moot point. However, one may argue that the correlation between "a" and "b" may be preserved within the discretized values produced through application of the Algorithm Organic on the reduced dataset. Such possibility is under investigation to determine whether or not the preservation of properties by a reduced dataset is sensitive to the purpose (goal) of mining.

In addition, the application of the proposed methodology on a very large dataset is under investigation which includes the viability study of the signatures as prediction rules along with the horizontal and vertical reductions of the signatures.

REFERENCES

[1] J. Han and M. Kamber, "Data Mining: Concepts and Techniques," Morgan Kaufmann publishers, 2001.

[2] M. Vojnović, F. Xu, and J. Zhou, "Sampling Based Range Partition Methods for Big Data Analytics," Technical Report of MSR-TR-2012-18, Microsoft Corporation, Redmond, WA, March 2012.

[3] M., Meilă, "Comparing Clusterings by the Variation of Information," Learning Theory and Kernel Machines, B. Schölkopf and M. K. Warmuth (Eds.), Springer Lecture Notes in Computer Science, Volume 2777, 2003, pp. 173–187

[4] T. Bell, I. H. Witten, and J. G. Cleary, "Modeling for text compression," The ACM Journal of Computing Surveys, vol. 21, no. 4, Dec. 1989, pp.557-591.

[5] S. Mahmud, "An Improved Data Compression Method for General Data", International Journal of Scientific & Engineering Research, vol. 3, no. 3, March 2013, pp.1-4.

[6] J. H. Pujar and L. M. Kadlaskar, "A new Lossless Method of Image Compression and Decompression Using Hufman Coding Techniques," Journal of Theoretical and applied Information Technology, vol. 15, no. 1, May 2010, pp. 18-23.

[7] D. Taubman, "High performance scalable image compression with EBCOT," IEEE Transactions on Image Processing, vol. 9, no. 7, July 2000, pp.1158-1170.

[8] Y. S. Wang, C. Wang, T. Y. Lee, and K. L. Ma, "Feature-preserving volume data reduction and focus+context visualization," IEEE Transaction: Visualization and Computer Graphics, vol. 17, no. 2, Feb. 2011, pp. 171-181.

[9] E. R. Tufte, "The Visual Display of Quantitative Information," 2nd Ed, Graphic Press Publisher, May 2001.

[10] S. Kang, J. Lee, H. C. Kang, J. Shin, and Y. G. Shin, "Feature-preserving reduction of industrial volume data using gray level co-occurrence matrix texture analysis and mass-spring model," Journal of Electronic Imaging, vol. 23, no. 1, Feb. 2014, pp. 13-22.

[11] T.R. Jones, "Feature Preserving Smoothing of 3D Surface Scans," MS. Thesis, Computer Science Department, MIT, Sept. 2003.

[12] R. M. Gary, "Entropy and Information Theory," Springer, 1990.

[13] J. Natt, R. Hashemi, A. Bahrami, M. Bahar, N. Tyler, and J. Hodgson, "Predicting Future Climate Using Algae Sedimentation," The Ninth International Conference on Information Technology: New Generation (ITNG-2012), Sponsored by IEEE Computer Society, Las Vegas, Nevada, April 2012, pp. 560-565.

[14] L. Yu and H. Liu, "Efficient Feature Selection via Analysis of Relevance and Redundancy," The ACM Journal of Machine Learning Research, vol. 5, Dec. 2004, pp. 1205-1224.

[15] R. Hashemi, A. Tyler, and A. Bahrami, "Use of Rough Sets as a Data Mining Tool for Experimental Bio-Data," Computational Intelligence in Biomedicine and Bioinformatics: Current Trends and Applications, T. G. Smolinski, M. G. Milanova, and A. E. Hassanien, (Eds.), Springer-Verlag Publisher, June 2008, pp. 69-91.

[16] R. Hashemi, A. Bahrami, M. Smith, N. Tyler, M. Antonelli, and S. Clapp, "Discovery of Predictive Neighborly Rules from Neighborhood Systems," International Conference on Information and Knowledge Engineering (IKE'13), Las Vegas, Nevada, July 2013, pp. 119-125.

[17] J. R. Quinlan, "Induction of Decision Trees," Machine Learning", vol. 1, no. 1, 1986, pp. 81-106.

[18] R. Hashemi, L. LeBlanc, and B. Westgeest, "The Effects of Business Rules on the Transactional Association Analysis," The 2004 International Conference on Information Technology: Coding and Computing (ITCC-2004), Pradip K. Srimani (Editor), Sponsored by IEEE, Las Vegas, Nevada, vol. II, April 2004, pp. 198 - 202.

[19] M. R. Chmielewski, and J. W. Grzymala-Bussse, "Global discretization of continuous attributes as preprocessing for machine learning, vol. 15, no. 4, Nov. 1996, pp. 319-331.

[20] J. Zhao and Y. H. Zhou, "New heuristic method for data discretization based on rough set theory," The journal of China Universities of Posts and Telecommunications, vol. 16, no. 6, Dec. 2009, pp. 113-120.

[21] Z. Pawlak, J. W. Grzymala-Busse, R. Slowinski, and W. Ziarko, "Rough Sets", Communications of ACM, vol 38, no. 11, Nov. 1995, pp. 88-95.

[22] R. Hashemi, B. Pearce, R. Arani, W. Hinson, and M. Paule. "A Fusion of Rough Sets, Modified Rough Sets, & Genetic Algorithms for Hybrid Diagnostic Systems", In: Lin TY, Cercone N Editors. Rough Sets & Data Mining: Analysis of Imprecise Data. Kluwer Academic Publishers, 1997. pp. 149-176.

[23] R. Hashemi, F. Choobineh, W. Slikker, and M. Paule, "A Rough-Fuzzy Classifier for Database Mining", The International Journal of Smart Engineering System Design, no. 4, 2002, pp. 107-114.

 ISBN: 978-1-61208-358-2