Attribute and Instance Based Data Reduction

[1]Harshalkumar Ramdas Khairnar, [2]Dr.Prashant M. Yawalkar
Computer Engineering, MET's,BKC, IOE, Nashik, Maharashtra, India.
[1]harshalkhairnar27@gmail.com, [2]prashant25yawalkar@gmail.com

***Abstract*—The data reduction techniques are preprocessing techniques. These techniques remove unimportant, less important, noisy, repetitive, and outlier data from the dataset and create representative data without changing the distribution of the dataset. The data is reduced in terms of instances and attributes. The data reduction helps to reduce the processing overhead of machine learning algorithms in terms of time and memory. It also requires less storage space. The attribute and instance reduction techniques are studied independently.**

The proposed approach presents a combine study of attribute and instance reduction. An attributes are removed using feature selection technique. The supervised pearson co-relation coefficient technique is used for selecting the important features. For instance reduction, Fast Data Reduction With Granulation Based Instances Importance Labeling (FDR-GIIL) technique is used. This technique uses Hausdorff distance and data crowding degree to mark an importance of label. The less important labels are removed from the dataset. The system results will be collected for UCI repository datasets. The system performance will be calculated in terms of execution time and accuracy of KNN classifier.

Keywords—data reduction, attribute reduction, instance reduction, filter, wrapper, granulation, instance labeling

I. INTRODUCTION

A lot of data is generated in a variety of fields like industries, organizations, scientific domains, social sites, etc. The explosive growth in data requires high storage space and processing power. Big data processing techniques are required to process such a large amount of data. Big data suffers from memory issues. Big data requires high storage and processing memory which is not available in every case. The machine learning algorithms are also hampered in terms of accuracy while processing such big data[2][3]. Some techniques are applied as a preprocessing technique to reduce the size of data. From this large amount of data, important data is filtered and representative data is generated. The important data is captured before applying machine learning algorithms. The data is filtered in terms of an instances and attributes. This is called a data reduction process. The instances are nothing but the number of records. Attributes represent the features of data.

In data reduction process noisy data, repetitive data, unimportant and less important data is removed. The data is removed without affecting the original data distribution. The data reduction helps to reduce the processing overhead of machine learning algorithms in terms of time and memory. The reduced data size also requires less memory for storage.

The data reduction process is classified into two sections based on the process of execution: filter and wrapper methods.

1: Filter methods:

In filter method, the algorithm is designed to selects instances/attributes using a selection metric. For instance selection, cluster centers, or marginal points are used as representative data. In this method distance function is used for selecting certain instances rather than the accuracy data classification.

In attribute selection attribute importance is measured using various techniques such as correlation coefficient, SU, etc. the following figure represents the filter method mechanism.

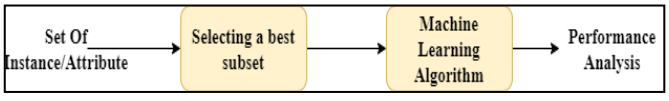

FIGURE 1 : FILTER METHOD

The filter methods are classified into 3 sections based on instance or attribute selection strategy[4].

1: Incremental approach: This approach begins with the empty set. One by one an attribute/instance is added to the set based

 DOI : 10.35291/2454-9150.2021.0351

on the algorithm result.

2: Decremental Approach: This approach removes the one by one unimportant attribute/instance from the set of whole data.

3: Batch approach: This approach analyzed all the attributes/ instances at ones in the first iteration and then decides which one needs to be preserved or removed.

4: Mixed Approach: This approach begins with training data i.e. set of attributes or instances are pre-selected and based on the training set new instances/attributes are added or deleted from the dataset.

2: Wrapper methods:

The wrapper methods are designed by focusing on accuracy parameters. The dataset classification is performed and attributes/instances that are not contributing or having fewer contributions are removed from the dataset[5].

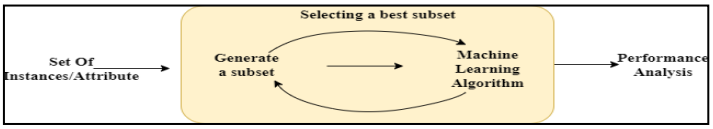

FIGURE 2 : WRAPPER METHOD

There is a need to make a tradeoff between classification accuracy, data reduction rate, and computational cost. The combined approach of filter and wrapper method tries to reach the accuracy as well as reduction rate by selecting the important instances from the data.

The techniques also differ as per the nature of the dataset. i.e. numerical dataset, categorical and hybrid dataset. The classification techniques and distance measures are changed with respect to the dataset.

The reduction methods are also classified based on the nature of the dataset. It can be classified as a supervised and unsupervised method. The supervised dataset contains labels for each instance. The whole data is categorized into two or more predefined classes. In an unsupervised dataset, no data labeling or categorization is present. The data reduction techniques are different as per the nature of the dataset.

In instance reduction, the unsupervised data is initially partitioned in multiple groups using the clustering technique and then the reduction is performed by removing the instances belonging to the same group and having the highest similarity with other instances. Whereas in supervised datasets the groups are already present in a dataset. The instance is removed by comparing instances within the same class.

In the attribute reduction method the relevance of attribute is compared with the class attribute for supervised datasets whereas the attribute relevance is compared with every other attribute in unsupervised dataset.

The instance reduction and attribute reduction methods play a vital role in filtering the datasets. These two techniques are studied independently. The combined approach will provide a great impact on data storage and machine learning algorithm processing.

The proposed approach works on the instance and attributes reduction technique. The supervised numerical dataset is used for data reduction. The correlation coefficient is used for attribute reduction. For instance reduction, a Granulation Based Importance Labeling technique is used.

In the following section work done in the data, reduction domain is discussed followed by problem formulation. The details of the proposed system are discussed in section IV, followed by Experimental setup and results. In the last section conclusion and future work is mentioned.

II. LITERATURE SURVEY

The work done in data reduction is mainly classified in following two sections:

A. Instance Reduction:

The instance reduction techniques are classified into the following two types:

1. Wrapper Methods:

Fast Condensed Nearest Neighbor (FCNN)[6] is the instance reduction technique based on the KNN classification accuracy. The instances are selected from the dataset based on the NN rule. It initially finds the centroid of classes. The centroid values and other dataset instances are compared using the Voronoi cell technique. The selected instances are added to the subset. This

 DOI : 10.35291/2454-9150.2021.0351

technique has sub-quadratic time complexity. The accuracy of classification depends on the order of selection. The technique requires user parameters for a number of instances count. This technique does not generate the smallest consistent subset. Further, some techniques are proposed to improve the working of CNN.

The generalized condensed nearest neighbor (GCNN)[7] technique selects the instances y comparing the nearest neighboring distance of an instance with respect to its nearest enemies. The enemies are nothing but the instances of different classes. The difference between the distances should be greater than the predefined threshold value P. GCNN produces the smaller subset of instances as compared to the FCNN. Determining the value of threshold P is a challenging task.

D. R. Wilson et. al. [8] proposes Decremental Reduction Optimization Procedures algorithms (DROPs). This is a collection of 5 algorithms: DROP1, DROP2,..,DROP5. The algorithm initially finds the associates of an instance. The system removes the instance if its associate instance is sufficient and plays the role of the selected instance in the classification process. The instance is get deleted if it does not affect the classification accuracy after deletion. The drop technique generates better results as compared to the CNN techniques. This algorithm is used as a noise filter algorithm before applying any machine learning algorithm. The DROP3 algorithm tries to manage a good tradeoff between instance reduction rate and classification accuracy as compared to the other DROPs techniques. But DROP3 has high computational complexity.

Based on the technique of local sets, E. Leyva et. al. [9] proposes a new technique. The technique has 3 variants: Local Set-based Smoother (LSSm), Local Set-based Centroids Selector method (LSCo) and Local Set Border selector (LSBo). In these methods, the set of instances created such that instances having the highest hypersphere centered on instances. hypersphere should not contain the instances from other classes. LSSm algorithm uses a decremental approach and tries to remove the instances from a dataset that are harmful to the classification process. The instances that generate false-positive results are removed from the dataset. LSCo initially applies LSSm for noise removal and then applies LS-clustering[11]. Using clustering results, only centroids are preserved. LSBo is also initially applying LSSm and then finds the local sets for each instance and instances are removed if it finds one or more representative instances in the local set. The complexity of each algorithm is O(N2). LSBo generates better results than LSSm and LSCo in terms of accuracy and reduction rate.

2. Filter Methods:

A. Lumini et al[12] proposed a filtering-based reduction technique based on clustering (CLU). The data is get divided into multiple clusters and the cluster centroid is treated as a representative data instance. J. A. Olvera-Lopez et. al.[13] proposes a supervised technique named Prototype Selection Based Clustering (PSC) algorithm. This technique tries to preserve the covariance structure of the class. It also uses a clustering technique same as CLU but the clusters are performed within class instances. In PSC, some inner instances of clusters are deleted but all boundary instances are preserved.

C. G. Vallejo et. al.[14] proposed a ranking-based technique called instance selection based on ranking (ISR). This is a supervised approach in which the training dataset is given as an input. The correlation of each instance with the training dataset is compared and the important instances are obtained by ranking. The attributes are arranged in decreasing order of their correlation value. The top k instances are selected as important instances. P. Hernandez-Leal et. al.[15] also proposes a ranking-based technique called borders for instance selection (IRB). This technique uses Edited Nearest Neighbor (ENN) algorithm to remove noise in the dataset. This technique then sorts the classified instances from boundary to the center The data on the boundary is preserved. Mainly the intraclass boundary instances are preserved and some of the inner instances are deleted.

J. L. Carbonera et. al.[16] proposed a local density-based instance selection (LDIS). This algorithm finds the density of each instance and preserves the instances with the highest density. The density of each instance is obtained by the neighboring instances of the same class. The whole dataset is divided into multiple granules and hence the complexity of the algorithm is less as compared to other algorithms that find the neighboring instances from the whole dataset. This technique is also applicable for big datasets.

Sun Xiaoyan et. al.[1] proposes a filter based Fast Data Reduction With Granulation Based Instances Importance Labeling technique. In this technique, data mapping[17] is performed. The mapping helps to reduce the dimension space and improve the efficiency of the reduction process. This technique follows the “divide and conquer” strategy. The data is divided into small granules using a clustering process. The instances are then compared with the same cluster instances. This reduces the processing overhead. For instance removal, Hausdorff distance, and crowding degree measurement techniques are used. Instances having less contribution and higher crowding degree are removed. This "divide and conquer" strategy helps to reduce computational cost and increases the efficiency of the reduction process. Hamidzadeh et. al.[18] proposes a survey on instance reduction techniques. The other older techniques for instance reduction and its details are discussed in this paper.

The filter and wrapper methods are studied independently. In the wrapper method, the data classification accuracy is the main criteria for reduction set evaluation whereas in filter methods some statistical formulae such as the distance measure and ranking are used. As wrapper methods use machine-learning algorithm for data reduction, the wrapper methods are more

 DOI : 10.35291/2454-9150.2021.0351

complex and time consuming as compared to the filter methods.

B. Attribute reduction:

U.M. Kaire et. al.[19] proposed a review on feature selection techniques The author mentions that the feature selection technique should provide good accuracy for classification as well as stability. Stability refers to the insensitivity of the algorithm after a small perturbation in data. In this paper, various filter-based, wrapper-based techniques are discussed. The feature selection strategies are mainly classified based on the algorithmic approach such as forward sequential search, backward sequential search, and hill-climbing.

Se-Hyun Ji et. al[20] proposed a correlation-based feature selection technique for Predicting the Bitcoin Transaction Count. Appropriate learning features are extracted from the whole dataset using the correlation coefficient technique.

III. PROBLEM FORMULATION

A lot of data is generated in a variety of applications. Data reduction is required to manage such data. The data reduction is useful for efficient usage of storage devices. This technique also improves the performance of the machine learning algorithm. The reduction techniques remove noisy data, repetitive data, unimportant and less important data from the dataset.

The reduction of the dataset is categorized in the Attribute reduction and Instance reduction process. The reduction is also classified in 2 section based on the algorithmic execution strategy of data reduction: Filter and Wrapper. These types of data reductions are always executed separately.

The efficiency of the reduction algorithm is as important as the accuracy of the reduction process. Various techniques try to manage the tradeoff between classification accuracy, data reduction rate, and computational cost. The collective study and implementation of these techniques help to reduce data size and improves the efficiency of the machine learning algorithm.

IV. PROPOSED APPROACH

A. Preliminaries:

1. Data Mapping:

Let dataset D{d_1,d_2,..,d_n} with n records having c+1 attributes as { d_{i1},d_{i2},..,d_{im}, c_i }. C_i is the class attribute. The mapping of m attribute to one attribute for each record T(d_i) is calculated using following formula:

$$T(d_i)= \sqrt{\sum_{j=1}^{c} d_{ij}^{\,2}}$$

2. Mapping Distance:

The distance between two instances d_i and d_j are calculated using mapping values of instances T(d_i) and T(d_j). It is calculated using following formula:

$$D_{ij} = T(d_i) - T(d_j)$$

$$D_{ij} = \sqrt{\sum_{i=1}^{c} d_i^{\,2}} - \sqrt{\sum_{j=1}^{c} d_j^{\,2}}$$

3. Hausdorff distance:

Let D be the dataset, D/d_i is the dataset without d_i instance. The Hausdorff Distance between two datasets is calculated using following formula:

$H(D, D/d_i) = \max(h(D, D/d_i), h(D/d_i,D))$

Where ,

$h(D, D/d_i) = ||d_p - d_q||$, where d_p is the maximum value in D and D_q is minimum value in D/d_i

Data Significance:

The significance of a data instance is calculated for selected the instance for removal process. Let instance d_i is removed from the dataset then the importance of x is calculated using Hausdorff distance.

$Sig(d_i) = H(D, D/d_i)$

4. Crowding degree:

The crowding degree of data instance d_i is calculated for instance removal selection purpose. The higher value of crowding degree indicates that too many instances are present near the selected instance. The instance having higher crowding degree is less important and it can be removed. The Crowding degree is calculated using following formula:

$$CD(d_i) = \frac{1}{\sum_{d_j \varepsilon d_\mu} ||d_i - d_j||} \ , \ d_\mu = \{d_i \mid sig(d_i) <= \mu\}$$

 DOI : 10.35291/2454-9150.2021.0351

5. Correlation Coefficient :

This is an attribute selected method. The correlation value of each attribute is calculated with respect to the class value. Let a_{ij} is the attribute value belonging to the class j. m_j is the class value of instance i. The correlation coefficient is calculated with c class value using following formula:

$$P = \frac{\sum_{j=1}^{c}\sum_{i}^{n}(a_{ij}-m_{ij})}{\sqrt{\sum_{i}(a_i-m_{ij})^2}}$$

The value of P varies between [-1 to 1]. -1 and 1 are the extreme values represent the complete correlation where as value 0 represent the independent nature of attribute.

B. System Architecture:

The proposed approach is based on attribute and instance reduction techniques. Initially, the dataset is input into the system. The attributes of the dataset are reduced using the correlation coefficient technique. The generated dataset is input into the instance deduction process. The Fast Data Reduction With Granulation Based Instances Importance Labeling[1] is used for instance reduction.

The accuracy of the dataset is calculated using KNN classifier. The accuracy of the original dataset, a dataset with attribute reduction, a dataset with instance reduction, and a dataset with attribute and instance reduction is compared. The following figure shows the architecture of the system.

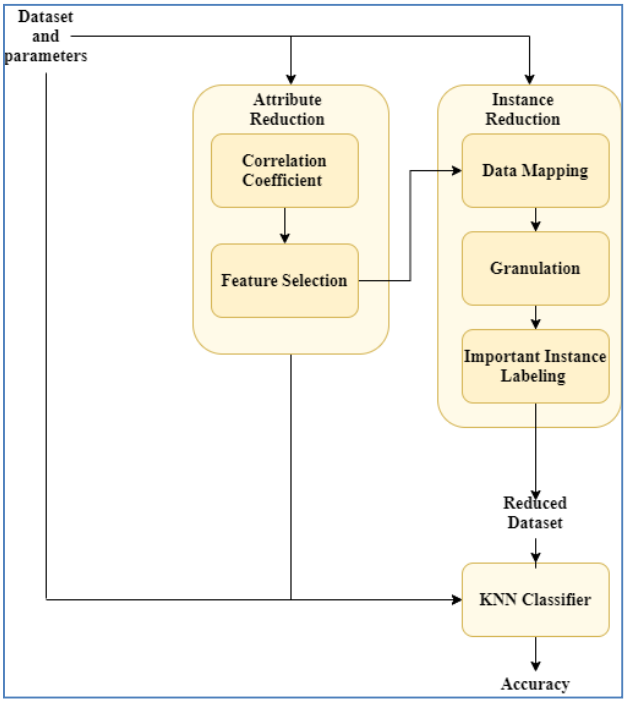

Fig 3: System architecture

C. System Working:

1. Attribute Reduction:

The process of attribute reduction follows the incremental approach. The important attributes are called a feature set. Initially feature set is initialized to an empty value. The attributes are added to the feature set based on its correlation coefficient value. The correlation coefficient value of each attribute is calculated with the class value and a number of attributes is selected based on the given attribute reduction rate. The correlation is calculated using formula mentioned in preliminary 6.

2. Instance Reduction:

Fast Data Reduction With Granulation Based Instances Importance Labeling technique (FDR-GIIL) algorithm is used for instance reduction. In FDR-GIIL working is classified into 3 sections:

1: Mapping: In mapping, the attributes of data are mapped to a one-dimensional array. The N attributes are mapped to one dimension. The mapping formula is mentioned in preliminary 6.

2: Granulation: In granulation the data is clustered in k granules. The granulation process divides the whole data in k sub granules. The granulation process is applied to improve the efficiency of the system. The data reduction process is executed for each granule. This reduces the computational cost. The k-means clustering algorithm is applied to the mapped dataset.

 DOI : 10.35291/2454-9150.2021.0351

3: Important instance labeling: For instance labeling significance value of each instance is calculated (mentioned in preliminary 4). For significance evaluation Hausdorff distance is used. Instances in granule are selected and its Hausdorff distance $H(D, D/d_i)$ is calculated. If Hausdorff distance is equal for two instances then the Crowding degree $CD(d_i)$ of those instances is calculated. The Hausdorff distance and Crowding degree formulae are mentioned in preliminary 3 and 5 respectively. The instance having higher crowding degree is less important and it can be removed.

The Important instance labeling process is executed iteratively until the data reduction size is reached. The detailed steps of proposed AIR algorithm are mentioned in the algorithm 1.

3. Accuracy Evaluation:

The accuracy of the dataset is evaluated using KNN classifier. The accuracy of the original dataset, the dimensionally reduced dataset and instance-based reduced dataset and attribute and instance-based reduced dataset accuracy is evaluated and compared for different reduction rate.

Algorithm: Attribute Instance Reduction (AIR) algorithm:

Input: D: Dataset
Ar: Attribute reduction rate
Ir: Instance reduction rate

Output: D': Attribute reduced dataset**.**
D'': Attribute-Instance reduced dataset**.**

Processing:

1. For each Attribute Ai in D
2. CC value: Calculate correlation coefficient
3. End For
4. Remove Ar % attributes from dataset D' based on CC value
5. For each Instance I in
6. Apply attribute mapping using Preliminary 1.
7. End For
8. K-Granule Creation: Apply Kmeans clustering algorithm
9. While Ir not reached
10. For all instances I in granule
11. Calculate Hausdorff distance (I) using preliminary 3
12. If Hausdorff distance (I)> μ
13. Retain the instance
14. End For
15. If two instances I and I' have same Hausdorff distance
16. Calculate the crowding degree if I and I' using preliminary 5
17. delete the instances with larger crowding degree
18. update dataset as D''
19. End While
20. Return D''

V. RESULT AND DISCUSSIONS

A. Experimental Setup:

For system implementation java language is used. The system is developed and tested on 4GB ram System with I7 processor. The jdk1.8 version is used for development. For classifier implementation weka 1.8 library is used. The graphical user interface is designed using swing component to interact with the system.

B. Datasets:

For system performance testing, various UCI datasets are used. The datasets are downloaded from UCI repository. The datasets are in text files. The datasets are initially preprocessed and converted in to ARFF(Attribute Relation File Format). Following table contains the list of datasets used.

 DOI : 10.35291/2454-9150.2021.0351

Table 1: Datasets

Sr. No.	Dataset	Number of attributes	Number of Instances
1.	Wine	13	178
2.	Zoo	17	101
3.	Magic	10	19020
4.	Hepatitis	19	155
5.	Segmentation	19	2100
6.	letter-recognition	16	20000

Algorithm Execution Time:

Following table contains the execution time details. The comparison of execution time between FDRGIIL algorithm[1] and proposed AIR algorithm is mentioned in the following table. The results are collected for three different reduction rates: 5%, 10% and 15 %. The time required for data reduction increases as we increase the reduction rate. In FDRGIIL algorithm only instance reduction is performed whereas in AIR algorithm instance and attribute reduction is proposed for data reduction. The AIR algorithm requires little higher time as compared to the FDRGIIL algorithm.

Table 2: Algorithm Execution Time

	Reduction rate 5%		Reduction rate 10%		Reduction rate 15%	
Dataset	FDRGIIL Execution Time (Reduction rate 5%)	AIR Execution Time (Reduction rate 5%)	FDRGIIL Execution Time (Reduction rate 10%)	AIR Execution Time (Reduction rate 10%)	FDRGIIL Execution Time (Reduction rate 15%)	AIR Execution Time (Reduction rate 15%)
Zoo	944	957	977	987	992	1065
Wine	1746	1810	1756	1886	1915	2145
Hepatitis	1485	1584	1625	1672	2106	2168
letter-recognition	6710	6792	6814	6981	7042	7146
Magic	16769	17212	17439	17848	18023	18274
Segmentation	3833	3970	6214	6512	9143	10051

The following figure shows the comparison of execution time between the FDRGIIL and AIR algorithms. The results are taken for 6 different datasets. The execution time is compared for three different reduction rates. The time required for execution depends on dataset size and the reduction rate. The time required for the AIR algorithm is higher than the FDRGIIL algorithm. AIR algorithm requires 2.8%, 3.5% and 4.8% higher time than FDRGIIL algorithm for reduction date 5%, 10% and 15% respectively. As we increase the reduction rate, the execution time required for both algorithms also increases. The X-axis represents the datasets whereas Y-axis represents the time required for execution.

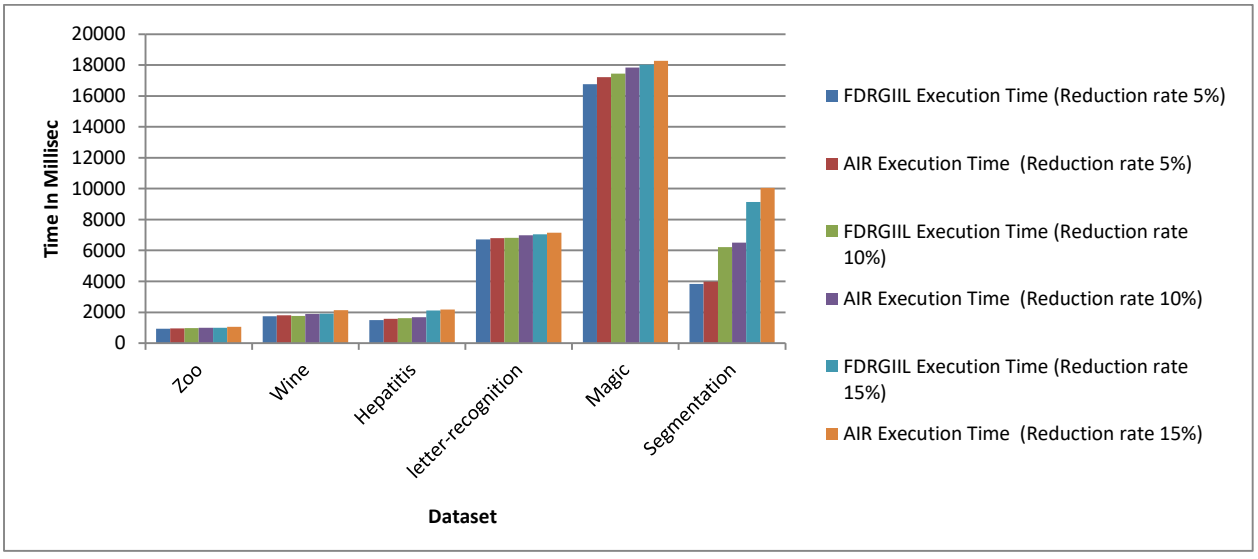

Fig 3: Algorithm Execution Time

 DOI : 10.35291/2454-9150.2021.0351

KNN Execution Time:

The KNN-3 algorithm is used as a classifier. The data reduction is performed on six different datasets with 3 different reduction rates as 5%, 10% and 15%. For higher reduction rate more data is get reduced and hence requires less time for classification. The data is reduced using FDRGIIL and AIR algorithm. The FDRGIIL algorithm only reduced the data in terms of instances whereas AIR algorithm reduced the data in terms of instances and attributes. The AIR algorithm requires less time as compared to the FDRGIIL algorithm.

Table 3: KNN Execution Time

		Reduction rate 5%		Reduction rate 10%		Reduction rate 15%	
dataset	KNN Runtime For Original Data	KNN Runtime For FDRGIIL Data	KNN Runtime For AIR Data	KNN Runtime For FDRGIIL Data	KNN Runtime For AIR Data	KNN Runtime For FDRGIIL Data	KNN Runtime For AIR Data
Zoo	752	743	708	729	693	702	559
Wine	945	785	674	777	659	764	652
Hepatitis	1032	995	952	871	854	761	547
letter-recognition	40112	37823	35287	36432	31967	32953	27300
Magic	29104	27826	26124	24097	24653	19824	15834
Segmentation	793	782	764	760	754	743	716

The following figure contains 6 different graphs as per the dataset. The x-axis represents the dataset and the y-axis represents the time in milliseconds. Each graph contain time required for classification for the original dataset, 5%reduced dataset, 10% reduced dataset and 15% respectively. FDRGIIL and Air algorithms are used for data reduction. The average reduced time of the FDRGIIL algorithm is 13.07% whereas the average reduced time for the AIR algorithm is 22.44%. The AIR algorithm improves the efficiency of the classification algorithm.

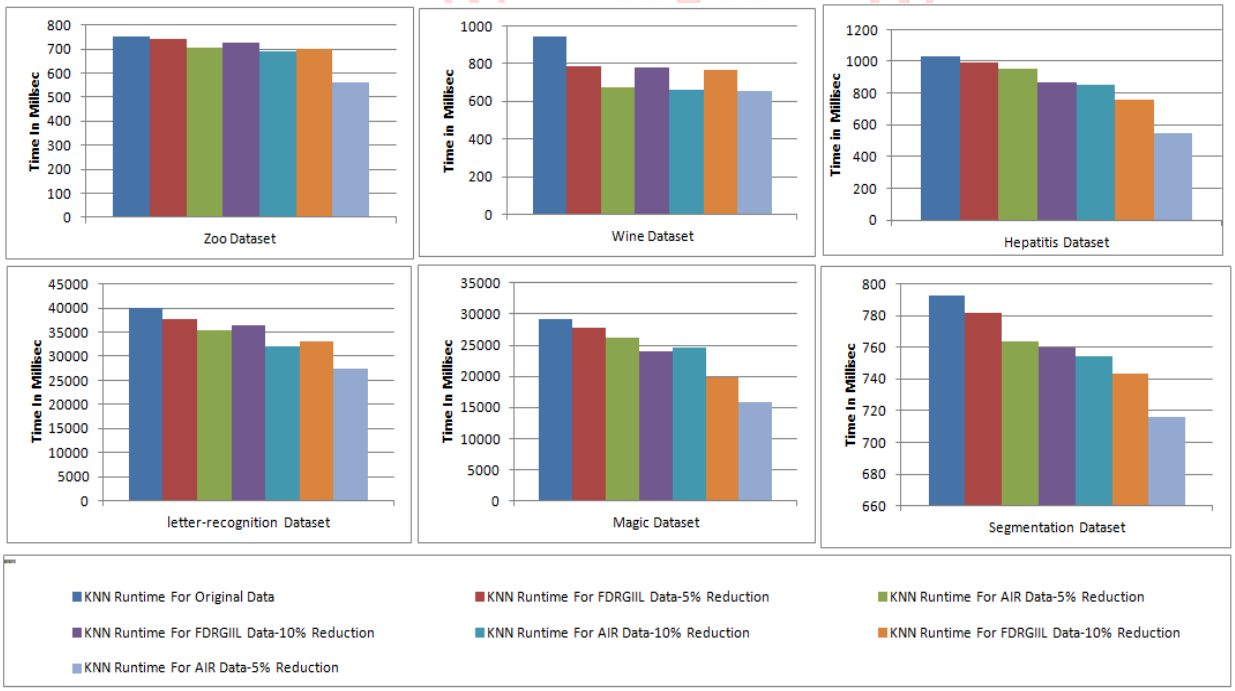

Fig 4: KNN Execution Time

KNN Accuracy:

The accuracy of KNN classifier is calculated for original and reduced dataset. The dataset is reduced using FDRGIIL and AIR algorithm. The reduction rate is defined as 5%, 10% and 15%. The accuracy of reduced dataset is nearly equal to the original dataset. The classification accuracy varies with respect to the dataset. The 5 %, 10% and 15% reduction of data do not affect the accuracy of classification.

 DOI : 10.35291/2454-9150.2021.0351

Table 4: KNN Accuracy

		Reduction rate 15%		Reduction rate 10%		Reduction rate 5%	
Dataset	Accuracy of original Dataset	FDRGIIL Accuracy	AIR Accuracy	FDRGIIL Accuracy	AIR Accuracy	FDRGIIL Accuracy	AIR Accuracy
Zoo	92.00	95.00	95.00	92.00	92.00	92.00	92.00
Wine	94.00	95.00	95.00	96.00	93.00	95.00	95.00
Hepatitis	80.00	79.00	78.00	77.00	80.00	76.00	80.00
letter-recognition	95.63	95.35	95.90	95.35	95.82	95.35	95.46
Magic	70.11	88.24	81.94	88.24	70.11	88.24	88.24
Segmentation	96.02	95.81	95.61	85.83	95.81	95.85	95.85

The following figure shows the KNN classification accuracy comparison of six different datasets. The X-axis represents the dataset and Y-axis represents the classification accuracy. The three graphs contain accuracy of 5% reduced dataset, 10% reduced dataset, and 15% reduced dataset respectively. The Classification accuracy varies with respect to the dataset. The classification accuracy of the reduced dataset is nearly equal to the original dataset.

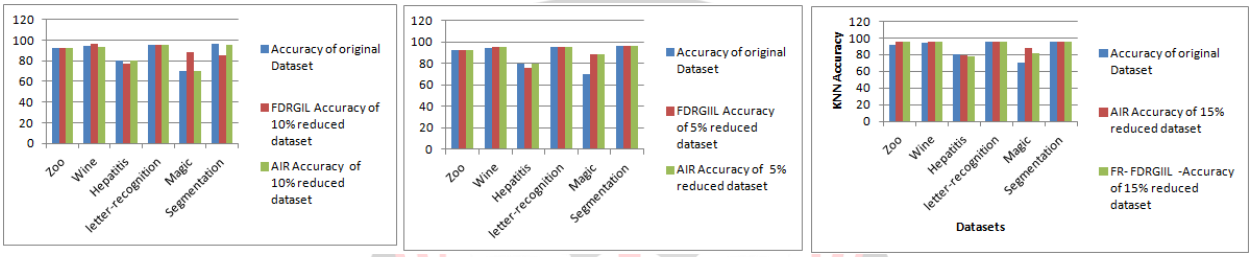

Fig 5: KNN Accuracy

VI. CONCLUSIONS

Data reduction is an important task. This is treated as the pre-processing task before applying machine learning algorithms. This technique reduces the data size and keeps only an important part of the data. The data can be reduced in terms of instances and attributes. The efficiency of the algorithm should be higher and the accuracy of the dataset should be nearly equal to the original dataset. Various techniques in the literature are trying to manage the tradeoff between data accuracy, efficiency and data reduction rate.

The combined approach of instance reduction and attribute reduction is proposed to generate a reduced data set and can be able to increase the efficiency of machine learning algorithms. The important attributes are filtered from an original dataset with the help of correlation coefficient and the attributes are removed using Fast Data Reduction With Granulation Based Instances Importance Labeling technique. The system results shows that after data reduction the classification algorithm efficiency increases and has nearly equal accuracy. The average reduced time of the FDRGIIL algorithm is 13.07% whereas the average reduced time for the AIR algorithm is 22.44%.

In future the system will be implemented for big data reduction on hadoop framework.

REFERENCES

[1] Sun, Xiaoyan & Liu, Lian & Geng, Cong & Yang, Shaofeng, "Fast Data Reduction with Granulation based Instances Importance Labeling", IEEE Access. PP. 1-1. 10.1109/ACCESS.2018.2889122.

[2] M. R. Bendre and V. R. Thool, "Analytics, challenges and applications in big data environment: A survey," J. Manage. Anal., vol. 3, no. 3, pp. 206-239, Jul. 2016.

[3] P. Guo, K. Wang, A-L. Luo, and M. Xue, "Computational intelligence for big data analysis: Current status and future prospect," J. Softw., vol. 26, no. 11, pp. 3010-3025, Nov. 2015.

[4] Arpita Joshi and Nurit Haspel (2020). A Novel Data Instance Reduction Technique using Linear Feature Reduction. Journal of Artificial Intelligence and Systems, 2, 191–206. https://doi.org/10.33969/AIS.2020.21012.

[5] S. Ougiaroglou and G. Evangelidis, "RHC: A non-parametric clusterbased data reduction for efficient k-NN classification," Pattern Anal. Appl.,ol. 19, no. 1, pp.

 DOI : 10.35291/2454-9150.2021.0351

93-109, Feb. 2016.

[6] Fabrizio Angiulli, "Fast condensed nearest neighbor rule", in Proceedings of the 22nd international conference on Machine learning (ICML '05). Association for Computing Machinery, New York, NY, USA, 25–32. 2005

[7] C.-H. Chou, B.-H. Kuo, and F. Chang, "The generalized condensed nearest neighbor rule as a data reduction method," in Proc. Int. Conf. Pattern Recognit., Hong Kong, pp. 556-559.],Aug. 2006.

[8] D. R.Wilson and T. R. Martinez, "Reduction techniques for instance-based learning algorithms," Mach. Learn., vol. 38, no. 3, pp. 257-286, 2000.

[9] E. Leyva, A. Gonzalez, and R. Perez, "Three new instance selection methods based on local sets: A comparative study with several approaches from a bi-objective perspective," Pattern Recognit., vol. 48, no. 4, pp. 1523-1537, Apr. 2015.

[10] M. Suganthi and V. Karunakaran, "Instance selection and feature extraction using cuttlefish optimization algorithm and principal component analysis using decision tree," Cluster Comput., vol. 1, no. 2, pp. 1-13, Jan. 2018.

[11] Caises, Y., Gonzalez, A., Leyva, E., P erez, R., "Combining instance selection methods based on data characterization: An approach to increase their effectiveness", Information Sciences 181(20), pp. 4780–4798, 2011

[12] Lumini and L. Nanni, "A clustering method for automatic biometric template selection," Pattern Recognit., vol. 39, no. 3, pp. 495-497, Mar. 2006.

[13] J. A. Olvera-López, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, "Anewfast prototype selection method based on clustering," Pattern Anal. Appl., vol. 13, no. 2, pp. 131-141, 2010.

[14] C. G. Vallejo, J. A. Troyano, and F. J. Ortega, "InstanceRank: Bringing order to datasets," Pattern Recognit. Lett., vol. 31, no. 2, pp. 133-142, Jan. 2010.

[15] P. Hernandez-Leal, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, and J. A. Olvera-Lopez, "InstanceRank based on borders for instance selection,"Pattern Recognit., vol. 46, no. 1, pp. 365-375, Jan. 2013.

[16] J. L. Carbonera and M. Abel, "A density-based approach for instance selection," in Proc. IEEE Int. Conf. Tools Artif. Intell., Vietri sul Mare,Italy, pp. 768-774,Nov. 2015.

[17] F. Zhang, B. Liu, and H. Yan, "Rough decision rules extraction and reduction based on granular computing," J. Commun., vol. 37, no. Z1, pp. 30-35, Oct. 2016.

[18] Hamidzadeh, J., Monsefi, R., Yazdi, H.S.: Irahc: Instance reduction algorithm using hyperrectangle clustering. Pattern Recognition 48(5), 1878–1889 (2015)

[19] Utkarsh Mahadeo Khaire, R. Dhanalakshmi, Stability of feature selection algorithm: A review, Journal of King Saud University - Computer and Information Sciences, 2019.

[20] Ji, Se-Hyun & Baek, Ui-Jun & Shin, Mu-Gon & Goo, Young-Hoon & Park, Jun-Sang & Kim, Myung-Sup, "Best Feature Selection using Correlation Analysis for Prediction of Bitcoin Transaction Count". 1-6. 10.23919/APNOMS.2019.8892896.

 DOI : 10.35291/2454-9150.2021.0351