Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

A Novel Clustering and Matrix Based Computation for Big Data Dimensionality Reduction and Classification

Jijo Varghese[1,*], P. Tamil Selvan[1]

[1] Department of CS, CA&IT, Karpagam Academy of Higher Education, Coimbatore, Tamil Nādu, India

ARTICLE INFO

Article history:
Received 12 April 2023
Received in revised form 3 June 2023
Accepted 13 August 2023
Available online 3 September 2023

Keywords:
Big Data; Clustering; Word Pattern; Similarity Measures; Dimensionality Reduction

ABSTRACT

For higher dimensional or "Big Data (BD)" clustering and classification, the dimensions of documents have to be considered. The overhead of classifying methods might also be reduced by resolving the volumetric issue of documents. However, the dimensions of the shortened collection of documents might potentially generate noise and abnormalities. Previous noise and abnormality information removal strategies include several different approaches that have already been established throughout time. To increase classification accuracy, current classifications or new classification methods that has created to conduct classification, must deal with some of the most difficult issues in BD document categorization and clustering. Hence, the goals of this research are derived from the issues that can be solved only by expanding classification accuracy of classifiers. Superior clusters may also be achieved by using effective "Dimensionality Reduction (DR)". As the first step in this research, we introduce a unique DR approach that preserves word frequency in the document collection, allowing the classification algorithm to obtain improved (or) at least equal classification levels of accuracy with a lower dimensionality set of documents. When clustering "Word Patterns (WPs)" during "WP Clustering (WPC)", we imply a new WP "Similarity Function (SF)" for "Similarity Computation (SC)" to be used as part of WPC. DR of the document collection is accomplished with the use of information gained from various WP clusters. Finally, we provide "Similarity Measures" for SC of high dimensional texts and deliver SF for document classification and deliver SF for document classification. With assessment criteria like "Information-Ratio for Dimension-Reduction", "Accuracy", and "Recall", we discovered that the proposed method WP paired with SC (WP-SC) scaled extremely effectively to higher dimensional "DataSets (DS)" and surpasses the current technique AFO-MKSVM. According to the findings, the WP-SC approach produced more favorable outcomes than the LDA-SVM and AFO-MKSVM approaches.

1. Introduction

There are different kinds of DS that contributes to the BD in its unique ways, but ultimately, the BD has been the sum of all these disparate streaming data in various formats and sizes [1]. Procurement of storage facilities in massive data warehouses and network-attached storage would

* *Corresponding author.*
E-mail address: jijo.22@gmail.com

https://doi.org/10.37934/araset.32.1.238251

be the major representation of BD's defining aspect "Volume". An extreme number of dimensions variation inside the DS is caused by the BD's huge volume, which in turn produces data diversity. Attempts to minimize the volume seem to be necessary for a proper analysis of BD [2].

In reality, DR refers to the procedure of transforming a high-dimensional DS together into low-dimensional DS where the same or comparable data could be communicated more efficiently also without altering the actual data. To maximize precision and minimize computing duration while also simplifying the output, DR could be viewed as a task of globally evolutionary computation with "Machine Learning (ML)", that decreases the number of characteristics, and eliminates inappropriate, noise, and data duplication. Numerous features which aggregate significantly further difficulties while evaluating the data thus facilitated the process of analysis extremely challenging once it arrives at BD [3].

Over the past three decades or more, data management approaches results showed themselves useful in several different areas of data analysis. Among these techniques, "Data Reduction (DaR)", is commonly employed in explanatory or descriptive research. Employing charts and other quantitative properties, it seeks to provide a concise overview of the collected data. The goal of multimodal exploration analytics would be to minimize the range of data dimensionality by identifying various factors such as "Components", "Clusters", etc. These account for the scatter in multivariate data, DaR comes firmly into this category [4].

DR techniques could be employed to create a synthetic perspective of data, simplifying the searching phase in circumstances when it is recognized that its data, which is sometimes large, includes valuable data. Data organization and information retrieval become the core of its challenge [5].

A combined variable's type, including "Quantitative" and "Qualitative", is common inside the environment of multidimensional DaR. Generally, data is processed either by transforming "Qualitative Variables" into "Quantitative Variables" for demonstration purposes or by separating "Quantitative Variables" into "Qualitative Variables". This frequently presents a distortion as a result of the decision to utilize a certain set of classes to determine whether or not its amplitudes are similar. This results in the decline of knowledge.

Under this setting, it's also common to encounter difficulties while discussing the reduction of diverse data due to the presence of both "Quantitative" and "Qualitative" characteristics within the same DS. In mathematics, the term "Quantitative Variables" is used to refer to those variables whose values can be expressed as a number [6].

Several scientists had developed approaches to this problem by turning these two kinds of variables into dynamic components. Techniques for limiting the conceptual range of data could be implemented by the selection or extraction of features. After being extracted, the features create a separate space that could be regarded as either "Linear" or "Non-Linear", depending on how they are combined. In contrast, attribute choice focuses on the best features based on some criteria [7].

Modern data is much more convoluted than historical data, hence DR is necessary. This one is extremely effective with the clustering algorithm since it reduces the overall size of the data while maintaining its exact analytical properties as the initial form. In an attempt to reduce the negative effects of the "Dimensionality Curse" and as well as accomplish a fast clustering computation, DR is required in every clustering method [8].

To perfectly manage those factors, properly deployed BD solutions might find a compromise between both the data management goals and also the data operational expense, which includes things like computing, administrative, and coding effort and time. Specialists in a wide variety of fields were capable of monitoring and gathering a massive quantity of data through advances in data collecting and database maintenance. Current data storage and analysis have significant problems,

moreover, when dealing with massive DSs [9]. A significant sticking issue throughout this direction would be a high range of features and sizes connected to a particular metric. The "Curse of Dimensionality" describes the challenge of the explosive growth in the complexity of doing a credible evaluation as the number of "dimension components" (which might include terms like "Variable", "Feature", and "Attribute") grows. Furthermore, the volume of data dimensionality tends to expand with extremely poor scaling of present methods. Data mapping from high-dimensional to low-dimensional form, with as much of the initial data architecture preserved as possible, becomes an approach to solving this issue [10].

The idea behind the proposed WP-SC approach is to cluster the WP incrementally.

- Initially, the number of clusters is zero. The process starts by initially assuming that the first "WPVector (WPV)" is the element of the first cluster. Initially, the cluster has only one element. So, the mean of the initial cluster is the first WP itself. Initially, any small value for deviation may be assumed as it cannot be equal to zero.
- Clusters are formed by considering each WPV and then determining its similarity to the existing clusters considering a user's threshold value. If the similarity constraint is satisfied, then the WPV is added to the existing cluster and the cluster information is updated to make sure that the resulting cluster generated has its mean updated. Similarly, in an event of failure of the user similarity condition and following which a new cluster is generated, the new cluster is updated accordingly. In this case, the mean of the cluster is the WPV which happened to be dissimilar considering existing clusters. The learning process is terminated when all WPVs are considered and clusters are generated.
- Once the WPC is done, the mean and deviation of the resulting clusters are computed and stored. This is now followed by the computation of similarities of WPVs to the generated clusters. The similarity of WPVs to each of the clusters that are generated is represented as the "Word Cluster Similarity Matrix". This matrix is known as the "Transformation Matrix (TM)".
- Thus, by applying the proposed method, the dimensionality-reduced text "Document Set (DoS)" can be obtained. The significance of the proposed method is that even after the feature transformation is performed, it is observed that the distributionof the words concerning"Text Documents (TD)" within dimensionality reduced DoS remains the same as original documents in the DoS.

Section 2 provides a synthesis of potentially relevant techniques concerning the research objectives of this study, Section 3 describes both current and proposed methods and techniques, Section 4 discusses exploratory performance comparison for both proposed and existing methods, and Section 5 concludesthe article with prospective future uses.

2. Related Works

The "K-Nearest Neighbor (KNN)" classification leverages a "Binary Search Tree (BST)" constructed analogously by the Hassanat *et al*., [11] to categorize BD. The key distinction would be as this approach relies on measuring a DS's dimension, which would be subsequently applied to organize the BST when retraining DS instances. This technique demonstrates the great capability for BD classification and could be extended to other contexts, especially wherever time is of the utmost importance. Once the created BST is organized by node, it is possible to find the farthest pairing of instances to every node. The KNN classification is applied to the instances throughout this study to determine how well they fit the test case. The outcomes demonstrated the technique's effectiveness

in accuracy and speed in comparison to the various techniques investigated. In contrast to traditional KNN, although, the accuracy levels were poor and could require some adjustment.

Hassanat [12] postulated using a KNN classification for sorting BD instances. After training where a BST could be explored within the polynomial duration, they combined the KNN classification also with the practice of putting training samples together into BST to accelerate the search. They evaluated 2 different approaches for classifying the training data. The "NBT" method would be the first, yet this takes the lowest and highest values of a metric and simplifies them down to zero or one, respectively. As opposed to the first technique, which puts every instance into the BST according to how close it appears to the maximum and minimum instances, this second method known as "MNBT", has been shown in investigations to be competitive in terms of classification accuracy and speed with the other techniques.

Gallego *et al*., [13] suggested a novel method that enhances the instance of the KNN technique, then the clustering methodology has been used to lower the computing expense of the KNN classifier. To acquire a precise description of the classification stage, "Neural Networks (NN)" was used to increase the training set's size, which significantly increases the efficiency and effectiveness. These suggested findings outperform the alternative approaches tested.

Since most techniques shouldn't scale effectively with greater levels of data dimensionality since several iterations develop substantially across most circumstances, Hassanat [14] designed four techniques to estimate the circumference of a multifaceted DS. The outcomes of tests run on several DS to demonstrate the effectiveness of the developed techniques. The above techniques are useful for calculating the circumference of a large DS, that has potential value in ML.

Pramanik *et al*., [15] performed comparative research that encompassed the most cutting-edge methods and software for BD categorization. Furthermore, they examine several methodologies currently being pursued in BD categorization. Various ML methods have been explored, along with their benefits and drawbacks. At last, a comprehensive overview of the different accessible frameworks employed in BD was shared with the public.

3. Research Methodology

In this proposed research methodology, some of the following preassumptions are considered to analyze both the existing and proposed models. Consider the "Document Set (DoS)" represented using Table 1. In the given DoS in Table 1 "d1, d2, d3, ..., dn" represent the TDs, and "w1, w2, w3, ... wm" are the words of the global word set. The first four TDs belong to "Class (c1)" and the last five TDs belong to "Class (c2)". The WPVs generated by considering DoS in Table 1 are represented using Table 2.

To explain the WPC process, a user dissimilarity threshold equal to 0.36 is considered, which means a similarity threshold is 0.64. So, the two WPs are considered similar WPs if and only if the similarity between them is at least 0.64. The working of the WPC process is shown in subsection 3.3.

3.1 LDA-SVM

The primary goal of the existing work is to investigate the impact of DR techniques on the efficiency of ML procedures. Each sector of the modern economy produces enormous amounts of data due to the intense competition throughout each field. To assist business owners and managers to generate more informed decisions, ML methods were employed to mine the data for meaningful information. DR methods have the potential tosimplify the time-consuming calculations required by the training phase drastically. In this research, the "Linear Discriminant Analysis (LDA)" is being

applied to retrieve the fundamental features of the DR system, while also being constructed to the widely used ML structured classification "Support Vector Machine (SVM)" through making use of publicly released DSs out from "UCI-ML library" [16].

Table 1
Sample DoS

	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	W_9	W_{10}	class
d_1	0	2	0	0	2	2	0	0	0	2	c_1
d_2	0	0	0	0	0	3	2	2	0	0	c_1
d_3	0	0	0	0	0	0	2	0	0	0	c_1
d_4	0	0	2	0	3	2	3	2	0	2	c_1
d_5	0	0	0	2	0	2	0	0	2	0	c_2
d_6	3	2	2	0	0	2	0	0	2	0	c_2
d_7	4	3	2	4	0	2	0	2	2	0	c_2
d_8	2	0	2	2	0	2	0	0	0	0	c_2
d_9	2	2	2	2	0	0	0	0	0	0	c_2

Table 2
WPVs

	Word patterns										
		X^1	**X^2**	**X^3**	**X^4**	**X^5**	**X^6**	**X^7**	**X^8**	**X^9**	**X^{10}**
Classes	**C_1**	0.00	0.22	0.20	0.00	1.00	0.47	1.00	0.67	0.00	1.00
	C_2	1.00	0.78	0.80	1.00	0.00	0.53	0.00	0.33	1.00	0.00

The following describes exactly LDA-SVM works:

- The first step of this approach entails applying LDA on individual DSs to determine which features are most important in predicting DR.
- Those features would then be sent into the SVM algorithm for training.
- One way to prepare information for further evaluation would be to acquire the reduced data using LDA's complete DS, calculate its variance, and afterward identify the occurrence of features.
- The process of selecting features from a DB that accurately represent relevant information of a DS would be a representation of the intrinsic dimensions of these databases.
- Standard statistical methods are used to compute the feature set by finding the spectrum of features that falls within a limit determined by the percentage variance of the DS. The typical range for the percentage of variation allowed is "97%" to "99%". By using these methods, it obtains intrinsic dimensionality between "3" and "6".

Whenever the feature set is narrow, it performs well. This means that the dimensions generated mostly by the LDA method are inadequate whenever a DS having considerably larger dimensions was selected, like those of "Hyper Spectral (HS)" data. A smaller intrinsic dimension might have been considered difficult due to the extremely high dimensions, as well as the chances of identifying most features could have been reduced.

In this case, raising the limit might improve the possibility of identifying relevant features and making correct recognition based on HS data. As a result, it's important to settle on an appropriate

limit. There is no way that it could be chosen at random. Examining to see whether the rule used to calculate the HS data's intrinsic dimensions has been modified. The LDA-SVM oriented DR strategy uses "Eigenvector" assessment, automatic selection of suitable transformed elements, and simultaneous categorization of HS data by employing "Non-Linearity SVM".

3.2 AFO-MKSVM

An "Adaptive Firefly Optimization (AFO)" approach that is centered on the "Map Reduce (MR)" framework has been created under this work. As a preliminary step, the "mapping stage" breaks down the massive DS into manageable segments called context blocks. Selecting features out of its big DS would then be done with the usages of the AFO method. Therefore, at completion, inside the "reduction stage," all the scattered information is consolidated as a unified feature set. The best features produced from AFO are therefore classified using the "Multi Kernel Support Vector Machine (MKSVM)" classification under this work for DR applications. This study's proposal is a unique AFO-MKSVM hybridization strategy that employs the MR transformation framework for feature selection; it is based on a meta-heuristic strategy. Both "Optimized FireFly" as well as "Adaptive Annealing" methods have been employed under this AFO to discover the best set of features. The fundamental purpose of this analysis becomes to enhance the accuracy of categorization simultaneously lowering the frequency of workflows [17,18].

The following describes exactly AFO-MKSVM works:

- During the initial stage, the massive DS is divided into manageable components called sample blocks.
- Although during the process of mapping, a deconstruction is carried out to better control the process of learning.
- The following stage is selecting features from massive DS entities using the AFO approach.
- Once the results of the reduction phase are combined, the MKSVM classifier checks out the refined features that have been generated.

3.3 WP-SC

3.3.1 WPC process

To explain the WPC process, a user similarity threshold equal to 0.64 is assumed. For a similarity threshold of 0.64, the deviation value is computed as 0.5388 by applying Equation (1). The value for "σ_l" is the same for all values of "l".

$$\sigma l = \frac{1-\theta sim}{\sqrt{ln\left(\frac{1}{abs(\theta sim)}\right)}} = \frac{0.34}{\sqrt{ln\left(\frac{1}{abs(0.64)}\right)}} = 0.5388 \quad (1)$$

Step-1: Initially, there are no clusters. The 1st cluster is formed by placing the first "WPV [X^1= (0.0, 1.0)]" into the first cluster. As this is the only WPV in the 1st cluster, therefore the mean of the 1st cluster is 1st "WPV [X^1= (0.0, 1.0)]". Let the 1st cluster be denoted as "Cluster-1". This means that "Cluster-1" has only the "WPV [X^1]".

Step-2: In the next step, the 2nd "WPV [X^2= (0.22, 0.78)]" is considered and the similarity of the 2nd WPV concerning the mean of the 1st cluster is determined. The similarity between the two WPVs is found to decide if the 2nd WPV can be added to the 1st cluster (or) if a new cluster has to be

created. The similarity between "X^1= (0.0, 1.0)" and "X^2= (0.22, 0.78)" is obtained. In this case, "Sim(X^i,X^j)" is equal to 0.9457. So, "X^2= (0.22, 0.78)" is added to the 1st cluster and the respective means are updated to "m^1= (0.11, 0.89)". Similarly, the 3rd "X^3"and 4th "X^4"WPVs are added to the 1st cluster and the mean is updated accordingly. At the end of Step-2, the mean is updated as "0.105, 0.895".

Step-3: In the next step, the 5th "WPV [X^5= (1, 0)]" is considered and the similarity of this WPV concerning the mean of the 1st cluster is determined. The similarity between "X^5= (1, 0)" and "m^1= (0.105, 0.895)" is obtained as nearly equal to 0 which is less than the allowable threshold for similarity. So, a new cluster is generated and the "WPV [X^5= (1, 0)]" is placed into the newly created cluster.

Step-4: In the next step, the 6th "WPV [X^6= (0.47, 0.53)]" is considered and the similarity of this WPV concerning the mean of the 1st cluster and 2nd cluster is determined. The similarity between "X^6"and the mean of 1st two clusters is less than 0.64. So, a new cluster is generated and the "WPV [X^6]"is placed into the newly created cluster. Thus, at the end of this step, 3 clusters are generated.

Step-5: Similarly, the remaining WPs are also clustered iteratively. Finally, three WPCs are generated. The mean and the standard deviation of all three clusters are represented in Table 3. Three "WPC G^1, G^2, G^3" are generated with "Cluster Sizes (4, 3, and 3)" respectively.

Table 3
Clusters Generated using the WPC process

Cluster	Size	Mean	Standard Deviation
G^1	4	(0.084,0.916)	(0.1152,0.1152)
G^2	3	(1,0)	(0,0)
G^3	3	(0.57,0.43)	(0.1414,0.1414)

Table 4 gives the final mean and final deviation of all three WPCs and cluster elements. The final deviation is obtained by adding the initial deviation based on which clusters are generated. Thus, three "Clusters G^1, G^2, G^3" are finally generated with "Cluster Sizes (5, 3, and 2)" respectively.

Table 4
Cluster Size, Mean and Final deviation of three clusters

Cluster	Elements	Size	Mean	Final deviation
G^1	**(X^1, X^2, X^3, X^4, X^9)**	5	(0.084,0.916)	(0.654,0.654)
G^2	**(X^5, X^7, X^{10})**	3	(1,0)	(0.5388,0.5388)
G^3	**(X^6, X^8)**	2	(0.57,0.43)	(0.6802,0.6802)

3.3.2 Transformation Matrix

This subsection explains the generation of TM. After the clusters are generated the similarities of WPVs to each cluster are determined. The similarity of WPs to three clusters is expressed as a "$[T]^{Soft}$"also called the soft TM. The "$[T]^{Soft}$" is represented by ten rows and three columns. The rows of "$[T]^{Soft}$" correspond to WP and the columns of "$[T]^{Soft}$" correspond to the clusters. An "Element Value (EV)" of the TM "$[T]^{Soft}$" represents the SV of the ith WP to the jth cluster. Figure 1 shows the Soft TM results.

$$[T]^{Soft} = [WC] = \begin{pmatrix} 1.0 & 0.0 & 0.25 \\ 0.92 & 0.02 & 0.59 \\ 0.94 & 0.01 & 0.55 \\ 0.97 & 0.0 & 0.25 \\ 0.02 & 1.0 & 0.45 \\ 0.50 & 0.14 & 0.96 \\ 0.02 & 1.0 & 0.45 \\ 0.20 & 0.47 & 0.96 \\ 0.97 & 0.0 & 0.25 \\ 0.02 & 1.0 & 0.45 \end{pmatrix}$$

Fig. 1. Soft TM Results

In soft TM, a WP may belong to one or more clusters. In hard TM, a WP belongs to only a single cluster. The hard TM can be obtained by setting each EV of the "$[T]^{Soft}$" matrix that has been equal or larger with the user's value of "1" as the threshold and setting the EV of the "$[T]^{Soft}$" matrix to "0" whenever the EV is less than user threshold.

$$[T]^{Hard} = [WC] = \begin{pmatrix} 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 1.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 1.0 \\ 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \end{pmatrix}$$

Fig. 2. Hard TM Results

In the present case, the user threshold is equal to "0.64". So, the EVs of the matrix "$[T]^{Soft}$" which are less than "0.64" are set to "0" while all other EVs of the "$[T]^{Soft}$" matrix are set to "1". The hard TM is hence obtained as depicted by "$[T]^{Hard}$". Figure 2 shows the Hard TM results.

3.3.3 Reduced dimensionality matrix

The final step is to obtain the DR representation of the original document set. The DR can be achieved by obtaining the product of a document matrix and hard or soft TM. Consider the DoS of Table 1 represented as a 2D matrix denoted by "$D^{original}$". Figure 3 shows the "$D^{original}$" results.

$$\mathbf{D^{original}} = \begin{pmatrix} 0 & 2 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 3 & 4 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 3 & 2 & 3 & 2 & 0 & 2 \\ 0 & 0 & 0 & 2 & 0 & 2 & 0 & 0 & 2 & 0 \\ 3 & 2 & 2 & 0 & 0 & 2 & 0 & 0 & 2 & 0 \\ 4 & 3 & 2 & 4 & 0 & 2 & 0 & 2 & 2 & 0 \\ 2 & 0 & 2 & 2 & 0 & 2 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Fig. 3. D original Results

Here, the hard TM is considered for demonstrating the DR. Thus, multiplying the original DoS with the hard TM, the DR representation of the original matrix is denoted by "$D^{Reduced}$". Figure 4 shows the "$D^{Reduced}$" results.

$$D^{Reduced} = \begin{pmatrix} 2 & 4 & 2 \\ 0 & 2 & 5 \\ 0 & 2 & 0 \\ 2 & 8 & 4 \\ 4 & 0 & 2 \\ 9 & 0 & 2 \\ 15 & 0 & 4 \\ 6 & 0 & 2 \\ 8 & 0 & 0 \end{pmatrix}$$

Fig. 4. D Reduced Results

It is visible from the "$D^{original}$" and "$D^{Reduced}$" matrices that the dimensionality of the original DoS set is "10" whereas the dimensionality of the DR representation of the original DoS set is equal to "3". It can easily be verified that the dimensionality of the initial DoS "$D^{Reduced}$" is reduced by 70%. With reduced dimensionality, the computational complexity of the learning algorithms can be greatly reduced and efficiency can be improved. The WP-SC method's advantage was " The document's word distribution is consistent both before and after DR".

3.3.4 Similarity measures for classifying text documents

After the DR of text DoS, the next step is to use the reduced DoS to perform text classification. Consider the reduced DoS set represented by "$D^{Reduced}$". Let the test document which must be classified is given by "$D^{(10)}$ = (4,3,3,1,1,3,1,1,1,1)".

The "Test Document Vector $D^{(10)}$" consists of ten dimensions. This document vector must be made dimensionality compatible considering other documents represented using reduced dimensionality text DoS represented using "$D^{Reduced}$". This can be achieved by multiplying the document vector with the hard TM. The "Transformed Document $D^{(10)}$" is given by "$D^{(10)}$ = (16, 5, 6)".

The similarities of the "Test Document $D^{(10)}$"to the TD represented using the "Reduced Document Matrix [$D^{Reduced}$]" and applying the "Similarity Measure (0.648, 0.676, 0.551, 0.603, 0.495, 0.520, 0.705, 0.498, 0.508". "Document $D^{(10)}$"has a maximum "Similarity Value (SV)" equal to 0.705 concerning $D^{(7)}$. Since the class label of "Document $D^{(10)}$"is C2. Hence, "Document $D^{(7)}$"is classified as "Class (c2)". The correctness of the classification result obtained using the DR's DoS can be verified by using the original DoS. Figure 5 demonstrates the SC between the "Test Document Vector" and the "Original TD" which are represented using "$D^{original}$".

It can be observed that the SC of the test document is maximum considering "Document $D^{(7)}$"which is equal to "0.61451". Since the "Document $D^{(10)}$" class-label was "Class (c2)". In the present case, we have obtained the "Document $D^{(7)}$"class-label as "Class (c2)".

	w1	w2	w3	w4	w5	w6	w7	w8	w9	w10	sim
d1	0	2	0	0	2	2	0	0	0	2	0.48772
d2	0	0	0	0	0	3	2	2	0	0	0.50699
d3	0	0	0	0	0	0	2	0	0	0	0.4202
d4	0	0	2	0	3	2	3	2	0	2	0.52654
d5	0	0	0	2	0	2	0	0	2	0	0.48761
d6	3	2	2	0	0	2	0	0	2	0	0.50477
d7	4	3	2	4	0	2	0	2	2	0	0.61451
d8	2	0	2	2	0	2	0	0	0	0	0.48098
d9	2	2	2	2	0	0	0	0	0	0	0.48837
std.dev	1.56347	1.22475	1.05409	1.45297	1.13039	1	1.20185	1	1	0.88192	
d10	4	3	3	1	1	3	1	1	1	1	

Fig. 5. SC of Test Document

4. Results and Discussion

The evaluating process is crucial because it allows for an in-depth examination of the collected data, revealing whether or not the composed data is accurate. Data for this assessment came from a wide range of sources within the study domain. The classified DS across the entire DB is evaluated based on those input characteristics, also known as real content. The studies are performed using the freely available "Cardiotocogram-Tracing (CT)" DS from the "UCI-ML repository", which is programmed using Java. A Windows-10 laptop featuring a RAM of 8GB is being used for this experiment.

Summary of the DS: Most pregnancies end up stressful for the mother throughout the third trimester.Oxygen delivery to the baby in the womb heartbeat is frequently interrupted at this time.By using CT, it is possible to monitor a developing child's heartbeat.It's also possible to track a baby's heart rate and uterine activity using this instrument.With a maximum of "2126 occurrences" including "23 features", this CT-DS is gathered from the "UCI-ML library". Major changes as from "Uterine Contractions per Second (UCs)" to "Fetal Movements per Second (FMs)".A baby's heartbeat could also be recognized by using several other features.Major features of the DS are outlined in Table 5.

Table 5
Significant features of DS

LB	FHR baseline (beats per minute)
AC	Accelerations per second
DL	Light decelerations per second
DS	Severe decelerations per second
DP	Prolongued decelerations per second
ASTV	percentage of time with abnormal short term variability
MSTV	mean value of short term variability
ALTV	percentage of time with abnormal long term variability
MLTV	mean value of long term variability

(i) Performance of DR

The fundamental objective of this investigation aims to compare the proposed WP-SC with the current LDA-SVM and AFO-MKSVM approaches by computing the "Information-Ratio (IR)" or DR, that reflects the true

features of the actual sources. Each dimension of the DB has been shrunk by a different amount, from 10 to 90, and the DR shifts accordingly. According to the results shown in Table 6 and Figure 6, the suggested WP-SC approach can get a strong DR in both higher and lower dimensions, whereas the DR for the current LDA-SVM and AFO-MKSVM was inferior.

Table 6
DR Performance

DIMENSION OF DATA	LDA-SVM	AFO-MKSVM	WP-SC
10	98.23	99.35	99.95
30	97.32	98.45	99.23
50	96.23	97.32	98.78
70	95.12	96.47	98.21
90	94.53	95.21	97.35

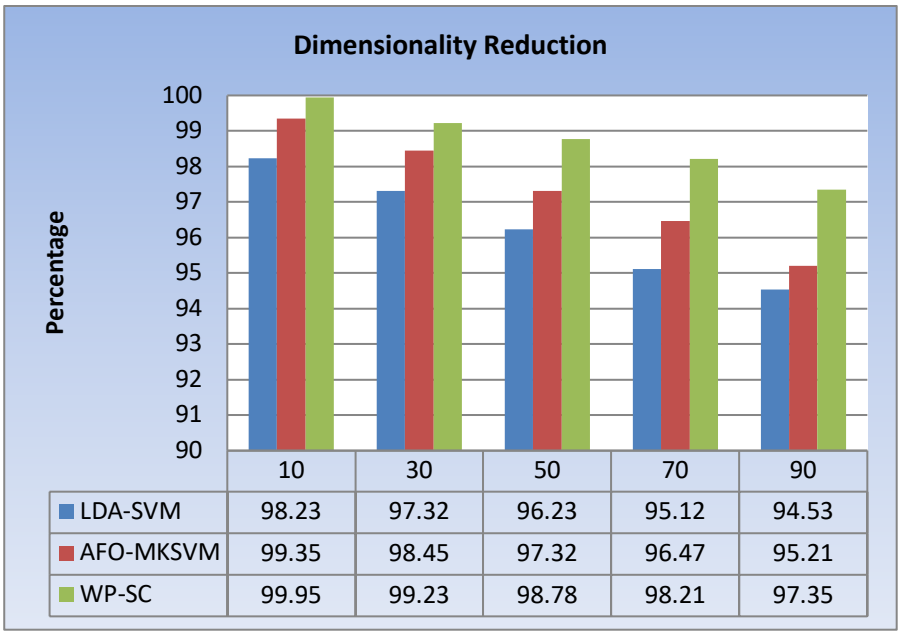

Fig. 6. DR Performance

(ii) Performance of Accuracy

Accuracy is evaluated using a "Confusion Matrix (CM)" that compares the raw data to the labeled DS.To calculate accuracy were using the following Equation (2):

"Accuracy = (True-Negative + True-Positive) / (True-Negative + True-Positive + False-Negative + False-Positive)**"** (2)

Every DS that has been placed into a category undergoes this process.The insights are often the result of an analysis of classified DS using data from the authentic domain.Extraction of features using the already available LDA-SVM and AFO-MKSVM techniques resulted in inferior accurate results of the classified DS, whereas selection of optimum features using the suggested WP-SC approach resulted in greater accuracy.It could be viewed as a positive result of the DR process. *A fundamental concept is shown by these processes:*Evidence found that its classifier's overall accuracy could be

increased by including TM for DRbychoosing merely the most promising features from the DS.Using the complete DB, Table 7 and Figure 7 compare the classification performance from WP-SC, AFO-MKSVM, and LDA-SVM with different DS and show their respective accuracy.

Table 7
Accuracy Performance

DS TYPES	LDA-SVM	AFO-MKSVM	WP-SC
DS 1	94.9	96.7	97.8
DS 2	96.3	98.3	99.4
DS 3	99.1	99.7	99.9
DS 4	98.4	99.2	99.9
DS 5	97.3	98.7	99.7

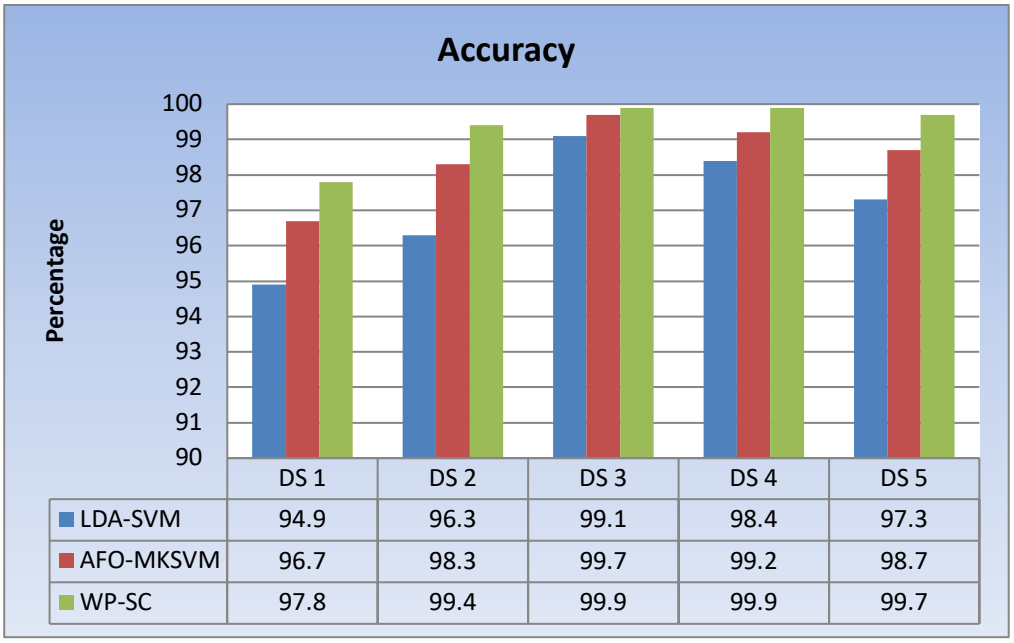

Fig. 7. Accuracy Performance

(iii) Performance of Recall

The ability to retrieve every element inside the DS having important features is measured by the Recall statistic. To calculate recall were using the following Equation (3):

"Recall = True-Positive / True-Positive + False-Negative**"** (3)

The metric "Recall" is used to describe the percentage of the total DS that contains correctly classified genuine data. All DSs that have been assigned a category undergo this process. The results are derived from an analysis of the classified DS with data from the authentic domain. Forming TM through optimum features using the suggested WP-SC approach improved its classified DS's recall over using solely extraction of features using the current LDA-SVM and AFO-MKSVM approaches. It can be considered a successful DR result as well. Table 8 and Figure 8 compare the results of the classification recall for the LDA-SVM, AFO-MKSVM, and WP-SC approaches using different DS within the entire DB.

Table 8
Recall Performance

DS TYPES	LDA-SVM	AFO-MKSVM	WP-SC
DS 1	93.8	94.9	96.2
DS 2	95.2	96.4	98.1
DS 3	98.2	99.3	99.9
DS 4	97.3	98.4	99.7
DS 5	96.2	97.5	98.9

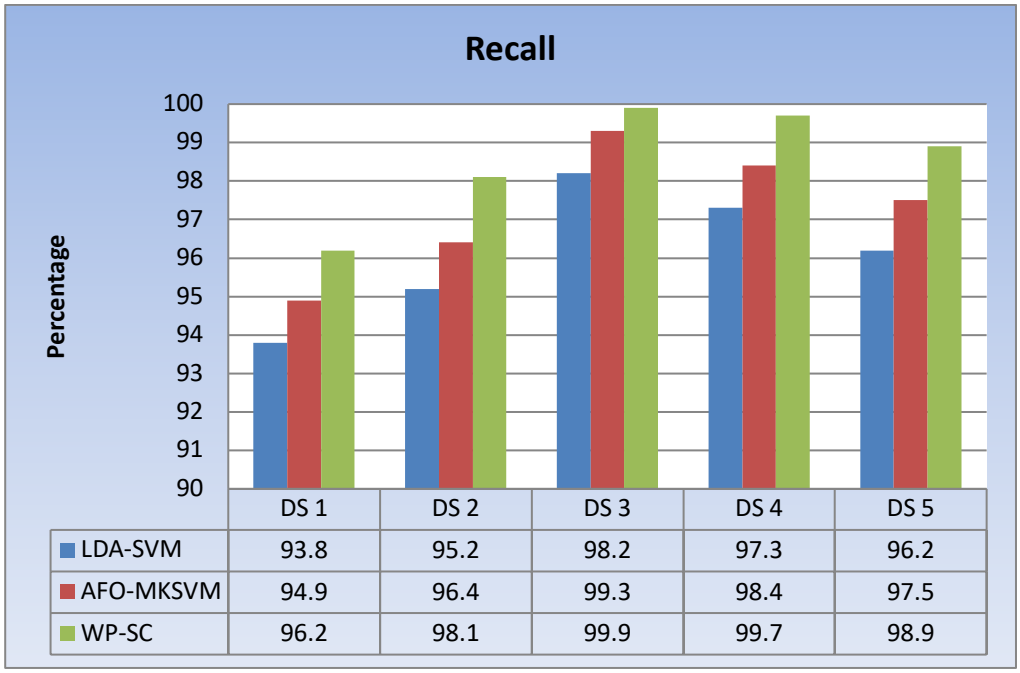

Fig. 8. Recall Performance

5. Conclusion

Many times, in real-time processing, the dimensionality of the document is usually very high and in some cases, it can even be sparse. Sparsity and high dimensionality of documents severe challenges in determining the similarity of documents to the text processing algorithm. The SCs are thus very crucial in classifying the documents. This research work proposed a novel WP-SC approach for SCamong WPVs.The proposed similarity function is used for SCamong 2 WPVs in the WPC process. This is followed by introducing a TM for DR whichimplies proposed similarity functions for SCs during WPC. The important advantage of the proposed approach is that the word distribution of TD in both the original and reduced DoS is the same before and completion of the DR. The proposed DR approach WP-SC performs superior to the existing LDA-SVM, and AFO-MKSVM approaches. In future, we implement and evaluate these models in large commercial datasets.

References

[1] Aragona, Biagio, and Rosanna De Rosa. "Big data in policy making." *Mathematical Population Studies* 26, no. 2 (2019): 107-113. https://doi.org/10.1080/08898480.2017.1418113

[2] Sun, Zhaohao, Lizhe Sun, and Kenneth Strang. "Big data analytics services for enhancing business intelligence." *Journal of Computer Information Systems* 58, no. 2 (2018): 162-169. https://doi.org/10.1080/08874417.2016.1220239

[3] L'heureux, Alexandra, Katarina Grolinger, Hany F. Elyamany, and Miriam AM Capretz. "Machine learning with big data: Challenges and approaches." *Ieee Access* 5 (2017): 7776-7797. https://doi.org/10.1109/ACCESS.2017.2696365
[4] Wang, Fang, Menggang Li, Yiduo Mei, and Wenrui Li. "Time series data mining: A case study with big data analytics approach." *IEEE Access* 8 (2020): 14322-14328. https://doi.org/10.1109/ACCESS.2020.2966553
[5] Ahmed, Nasim, Andre LC Barczak, Teo Susnjak, and Mohammed A. Rashid. "A comprehensive performance analysis of Apache Hadoop and Apache Spark for large scale data sets using HiBench." *Journal of Big Data* 7, no. 1 (2020): 1-18. https://doi.org/10.1186/s40537-020-00388-5
[6] Padillo, Francisco, José María Luna, and Sebastián Ventura. "Evaluating associative classification algorithms for Big Data." *Big Data Analytics* 4 (2019): 1-27. https://doi.org/10.1186/s41044-018-0039-7
[7] Raja, Rakesh, Indrajit Mukherjee, and Bikash Kanti Sarkar. "A systematic review of healthcare big data." *Scientific programming* 2020 (2020): 1-15. https://doi.org/10.1155/2020/5471849
[8] Du, Miao, Kun Wang, Zhuoqun Xia, and Yan Zhang. "Differential privacy preserving of training model in wireless big data with edge computing." *IEEE transactions on big data* 6, no. 2 (2018): 283-295. https://doi.org/10.1109/TBDATA.2018.2829886
[9] Jijo Varghese,and Dr. P. Tamil Selvan (2021, April). Machine Learning Techniques for Dimensionality Reduction in Big Data. Turkish Online Journal of Qualitative Inquiry (TOJQI), 12(2), 743–755. https://www.tojqi.net/index.php/journal/article/view/9320/6628.
[10] Khan, Z. Faizal, and Sultan Refa Alotaibi. "Applications of artificial intelligence and big data analytics in m-health: a healthcare system perspective." *Journal of healthcare engineering* 2020 (2020): 1-15. https://doi.org/10.1155/2020/8894694
[11] Hassanat, Ahmad BA. "Furthest-pair-based binary search tree for speeding big data classification using k-nearest neighbors." *Big Data* 6, no. 3 (2018): 225-235. https://doi.org/10.1089/big.2018.0064
[12] Hassanat, Ahmad BA. "Norm-based binary search trees for speeding up knn big data classification." *Computers* 7, no. 4 (2018): 54. https://doi.org/10.3390/computers7040054
[13] Gallego, Antonio-Javier, Jorge Calvo-Zaragoza, Jose J. Valero-Mas, and Juan R. Rico-Juan. "Clustering-based k-nearest neighbor classification for large-scale data with neural codes representation." *Pattern Recognition* 74 (2018): 531-543. https://doi.org/10.1016/j.patcog.2017.09.038
[14] Hassanat, Ahmad. "Greedy algorithms for approximating the diameter of machine learning datasets in multidimensional euclidean space: Experimental results." (2018). https://doi.org/10.14201/ADCAIJ2018731530
[15] Pramanik, Pijush Kanti Dutta, Saurabh Pal, Moutan Mukhopadhyay, and Simar Preet Singh. "Big Data classification: techniques and tools." In *Applications of Big Data in Healthcare*, pp. 1-43. Academic Press, 2021. https://doi.org/10.1016/B978-0-12-820203-6.00002-3
[16] Jijo Varghese, and Dr. P. Tamil Selvan (2022). A feature reduction based LDA withSVM classification on dimensionality reduction for Big Data. International Journal of HealthSciences, 6(S2), 9415–9431. https://doi.org/10.53730/ijhs.v6nS2.7461
[17] Varghese, J., and P. T. Selvan. "An adaptive firefly optimization (AFO) with multi-kernel SVM (MKSVM) classification for big data dimensionality reduction." *International Journal on Recent and Innovation Trends in Computing and Communication* 10, no. 7 (2022): 100-111. https://doi.org/10.17762/ijritcc.v10i7.5595
[18] Subburayalu, Gopalakrishnan, Hemanand Duraivelu, Arun Prasath Raveendran, Rajesh Arunachalam, Deepika Kongara, and Chitra Thangavel. "Cluster based malicious node detection system for mobile ad-hoc network using ANFIS classifier." *Journal of Applied Security Research* 18, no. 3 (2023): 402-420. https://doi.org/10.1080/19361610.2021.2002118