

# Medical Big Data Classification Using a Combination of Random Forest Classifier and K-Means Clustering

**R. Saravana kumar**

Professor, Department of computer science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore  
E-mail: saravanaram0516@gmail.com

**P. Manikandan**

Professor, Computer Science and Engineering Department from Malla Reddy Engineering College for Women, Maisammaguda, Secunderabad, Telangana  
E-mail: parasumani001@gmail.com

Received: 19 February 2018; Accepted: 20 May 2018; Published: 08 November 2018

**Abstract**—An efficient classification algorithm used recently in many big data applications is the Random forest classifier algorithm. Large complex data include patient record, medicine details, and staff data etc., comprises the medical big data. Such massive data is not easy to be classified and handled in an efficient manner. Because of less accuracy and there is a chance of data deletion and also data missing using traditional methods such as Linear Classifier K-Nearest Neighbor, Random Clustering K-Nearest Neighbor. Hence we adapt the Random Forest Classification using K-means clustering algorithm to overcome the complexity and accuracy issue. In this paper, at first the medical big data is partitioned into various clusters by utilizing k-means algorithm based upon some dimension. Then each cluster is classified by utilizing random forest classifier algorithm then it generating decision tree and it is classified based upon the specified criteria. When compared to the existing systems, the experimental results indicate that the proposed algorithm increases the data accuracy.

**Index Terms**—Decision trees, k-means clustering, medical big data, random forest, Classification.

## I. INTRODUCTION

Massive and complex data usually represented in exabyte ( $10^{18}$  bytes) are referred as big data. The large sensitive data is being used frequently in various organizations such as biomedical, IT, banking and so on. Using conventional database and other data analysis tools such data is difficult to manage, standardize [1] and secure [2, 3]. Regarding their structure, storage and analysis, medical big data involve many issues. Several techniques are to be followed [4-8] to increase the accuracy, cost reduction and improve the efficiency of big data. Big data uses the traditional classification

algorithms, such as decision trees, support vector machine, Naive Bayes neural network, and k Nearest Neighbors (kNN), Differential Evolution(DE)[9] algorithm, Machine Learning [10] algorithm, Big Data Analytics(BDA)[11, 12].

Fast finding the nearest samples and selecting representative or eliminating certain samples are the two traditional KNN methods utilized in big data. For training, the earlier version of SVM-KNN [13, 14] has to compute the query distances which are comparatively slow. However in big data the computational complexity is high. The task of partitioning a feature space into fuzzy classes is called the Fuzzy classification. In each region the feature space can be specified with fuzzy regions, which is maintained using fuzzy rules [15]. Feature subset selection and linear discriminate analysis are the two methods used in neuro-fuzzy classifier. These methods are used to evaluate the important feature subsets. Hence for training the neuro-fuzzy classifier, the characteristics of the data distribution is restored in feature space [16]. Usage of many fuzzy rules is the main drawback of this method. The fuzzy neural classifier algorithm resulted weak identification of data and hence cost and time is increased.

In this paper, to classify the medical big data, we propose a combined clustering and classification technique. The proposed technique is the joined execution of both the k-mean clustering method and RF (Random Forest) classification method. Compared to other clustering algorithm the k-means clustering gives better performance. Since, it takes less time for partitioning the high-dimensional datasets. The RF is an efficient learning method which is easy to interpret and explain non-parametric. At first, the k-mean clustering method is utilized to separate the high-dimensional medical data into various parts where each partition is considered as a cluster. The difference between each cluster member and the mean of the cluster value is

computed after the mean estimation of each cluster. Then, the clustered information is classified using the RF classification technique. To effectively recognize the underrepresented class, this RF technique can oversee datasets as vast as required giving the vital support. By the proposed RF approach the big medical data will be reduced accurately and efficiently.

The rest of the paper is organized as follows. Section 2 briefly reviews the related works. Section 3 elaborates the training and testing process. The proposed technique achievement results and the related discussion are given in section 4 and the paper is concluded in section 5.

## II. RELATED WORKS

Gang Luo [17] have introduced a system named Predict-ML for transformation of big clinical data into several datasets which was used in various applications. The results were predicted automatically, in which the main advantage of the system is less time and reduced cost. The Predictive model can guide personalized medicine and clinical decision making. The software takes several years to be built fully and hence not easily affordable.

For more flexibility in dynamic data streams a new evolving interval type-2 fuzzy rule-based classifier (eT2Class) was presented by Mahardhika Pratama *et al.*, [18]. While retaining more compact and parsimonious rule base on the state-of-art EFC's that method produces more reliable classification rates. Referring to the summarization and generalization power of data streams, initial data stream was pruned and the fuzzy rules were grown automatically. Accuracy and reliability were accomplished in that technique. However the complexity was prediction and management of big data.

A KNN rule classifier based on GPU devices was designed by P. Gutiérrez *et al.*, [19] to overcome the dependency between datasets and the GPU memory requirement. Using this method, an efficient CPU-GPU communication was designed. Due to its Irrespective size GPU keeps the memory usage stable and allows the dataset addressing significantly from hours to minutes in which the run time has been reduced. The design was best suitable in lazy learning algorithms such as KNN rule. The time complexity was reduced and also the run-time performance was improved. But for every training datasets the nearest distance calculation was complex for big data.

In order to reduce the complexity of big data a novel architecture was introduced by Entesar Althagafy and M. Rizwan Jameel Qureshi [15]. To efficiently analyze, the big data in IT companies includes large complex data's were difficult. To improve the performance of quality of service (QoS) this system integrates the Amazon Web Service (AWS) remote cloud, Eucalyptus, Hadoop. All incoming requests were accepted by AWS remote cloud and to the best proper Eucalyptus cloud it is forwarded intelligently. The complexity was overcome by this architecture and the performance against incoming request from multiple clients was enhanced and hence the

time and cost complexity was reduced and the QoS was increased. Data security and privacy was the major limitation.

Zhang Yaoxue *et al.*, [20] have presented the survey on cloud computing and analyzed its related distributed computing technologies. To support the big data of IoT, two promising computing paradigms were introduced they are transparent computing and fog computing. The computational performance against big incoming data requests was improved from multiple clients but there was less sensitive data security.

A big data analytics-enabled business value was introduced by Yichuan Wang and Nick Hajli [21]. All the phases the information life cycle in big data architecture was made understood by the concept of ILM. By analyzing secondary data consisting of big data cases specifically in the healthcare context it also explores the three path-to-value chains to reach big data analytics success. Thus to analyze big data for business transformation it provides new methodology to healthcare practitioners and detailed investigation of big data analytics implementation was offered more. There was no proper method to retain and manage data efficiently, though it has flexibility to deal with big data.

To reduce the challenges faced by big data in radiology and other healthcare organization P. Marcheschi [22] implemented HL7 (High Level 7) CDA technique. Standard radiology was highly benefitted due to the presence of DICOM (Digital Image and Communication in Medicine) and faster implementation can be done by the developers. The dissemination usage of FHIR standard simplifies developer works to make it less abstract, for the process of document creation. The implementation simplifies the presence of more templates and for its simplicity and completeness it was hence proved to be more successful. Lack of "plug and play" solution that helps in the standardization of data was the major drawback.

The devices that track real-time health data, or devices that auto-administer therapies, devices that constantly monitor health indicator when a patient self-oversees a therapy, D. Dimitrov [23] initiated mIOT (medical Internet of Things) and big data. In smart phones, wireless devices can be implemented and the time spends by the end users can be reduced by that method. Based upon the symptoms it can be diagnosed. However it cannot determine the exact health condition of users this method cannot be fully trusted. Accordingly modifications were done by updating current data and also it should be made user friendly.

## III. PROPOSED METHOD

Training process and testing process are the two processes being introduced in this section. In training process, initially the medical big data is partitioned using K-means clustering. Then randomly select the partitioned datasets and for each dataset, decision trees are generated. In testing process, each test sample is classified from the values generated in decision trees.

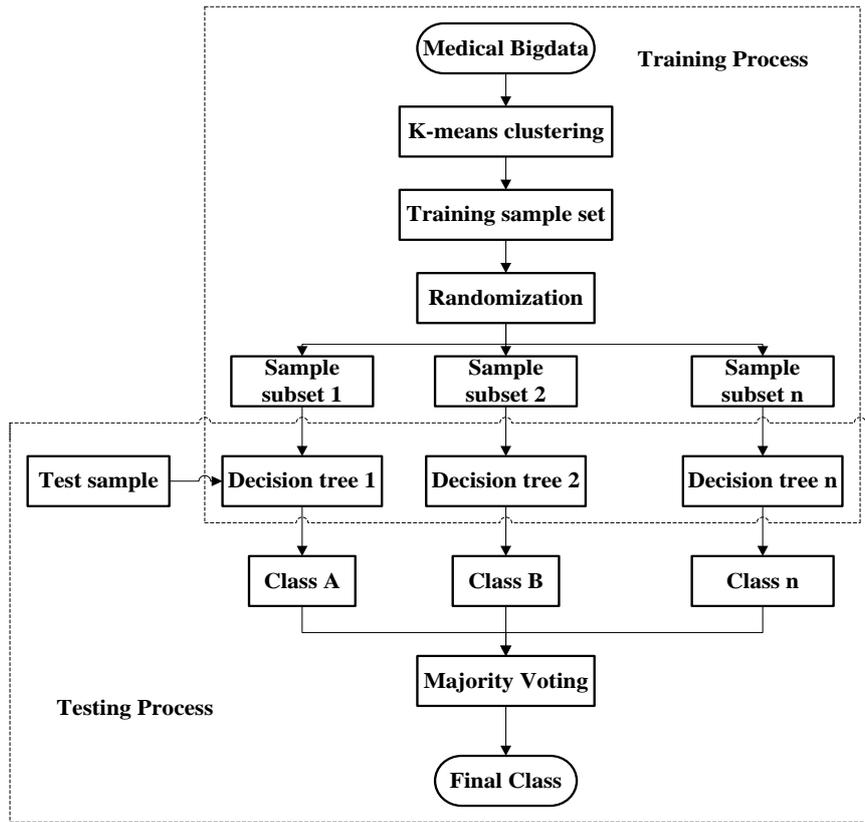


Fig.1. Random forest Classification using  $K$ -means Clustering

The classification of medical big data using random forest is shown in fig. 1. By using k-means clustering the medical big data is partitioned into number of groups named as clusters. Then, based on the RF classification method the cluster data is classified and which are start with decision tree generation. For each split, random forest selects random subset of predictors. Even on smaller sample set sizes. Random selection further reduces variance and hence the accuracy is increased. Here, for each decision tree the random values are generated and finally the best class is selected based on majority voting.

A. Training Process

The medical big data contains massive and complex data such as management information, staff details, data and medicinal information. Random forest classifier is used to classify such data accurately and efficiently. To partition the big data at first training process clustering is done for selection of random subsets. The decision trees are generated from the random subsets.

B.  $K$ -means clustering

1) *Choosing  $K$* : In  $K$ -means clustering to produce the best result the repeated refinement is used.  $K$  is given as input for the dataset and the number of clusters. The clusters are determined by the  $K$  means algorithm and for a particular pre-chosen  $K$  data set labels are determined. The number of clusters in the datasets for a range of  $K$  values is found using the  $K$ -means clustering algorithm and hence the

results are compared. Generally, for determining exact value of  $K$  there is no method, but using certain methods an accurate estimate can be obtained. Across different values of  $K$ , the mean distance between data points and their cluster centroid, the most commonly used method is to compare their results. Whenever the number of clusters is increased the distance to data points is reduced, and will always decrease this metric hence increasing  $K$  and when  $K$  is the same as the number of data points it goes to the extreme of reaching zero.  $K$  can also be determined using certain methods like Cross-validation, information criteria, the information theoretic jump method, the silhouette method, and the G-means algorithm and so on. Across a group that provides insight into how the algorithm is splitting the data for each  $K$  the distribution of data points can be monitored. In the data point the data set is a collection of features. Either randomly generated or selected from the data set. Initially the algorithm starts with selection of cluster centroid. Between two steps the algorithm then iterates:

*Step 1: Data assignment:* Each cluster from the dataset is defined by cluster centroid. In data assignment step, each data point is assigned to the centroid with the minimum distance based on the squared Euclidean distance. More formally, in set  $C$  if  $C_i$  is the collection of centroids, then each data point  $x$  is assigned to a cluster based on

$$\arg \min_{c_i \in C} \text{dist}(c_i, x)^2 \quad (1)$$

Here, in equation 1, using Euclidean distance the minimum distance between dataset and the centroid is calculated. For each  $i^{\text{th}}$  cluster centroid the set of data point assignments be  $S_i$ .

*Step 2: Centroid updation:* The cluster centroid are recomputed. In equation 2, this is done by taking the mean of all data points assigned to that centroid's cluster.

$$c_i = \frac{1}{|S_i|} \sum_{x_i \in S_i} x_i \quad (2)$$

The two steps are repeated until no data points change clusters. To converge a result this algorithm is guaranteed. The possible outcome is not produced by the result, meaning that with randomized starting centroids may give a best output assessing more than one run of the algorithm.

### C. Random Forest Algorithm

*Input:* Training Datasets, Test sample

*Output:* Majority vote from all individual trained trees after classification.

Let  $T_{\text{trees}}$  be the number of trees to build.

For each of  $T_{\text{trees}}$  iterations

1. From training set a new bootstrap sample is to be selected.
2. On the bootstrap an unpruned tree is grown.
3. Randomly select  $m_{\text{try}}$  at each internal node predictors and using only these predictors determine the best split.

The datasets best split is chosen based upon the regression and classification of data. Efficiently  $T_{\text{trees}}$  is selected by building trees until entire dataset is splitted. For better split,  $m_{\text{try}}$  is the number of predictions and it is randomly selected from the dataset. Where,  $m_{\text{try}} = k$  bagging is a special case in random forest. Many benefits of decision trees such as handling missing values, continuous and categorical prediction are retained by Random forest. The forest model is built initially. To make predictions we use the forest. In general the random forest can be specified as follows.

*Step 1:* At first the dataset is to be created as,

$$S = \begin{bmatrix} fx1 & fy1 & \dots & Z1 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ fxn & fyn & \dots & Zn \end{bmatrix} \quad (3)$$

Where  $fx1$  is the feature  $x$  of 1<sup>st</sup> sample,  $fyn$  is the feature  $y$  of  $n^{\text{th}}$  sample. Likewise  $Z_{\text{th}}$  feature samples are determined. The randomized training dataset is depicted in equation 3.

*Step 2:* Next based upon the criteria the random data subsets are created. These subsets are called as decision trees.

Decision tree1

$$S1 = \begin{bmatrix} fx12 & fy12 & \dots & Z12 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ fx35 & fy35 & \dots & Z35 \end{bmatrix} \quad (4)$$

Decision tree2,

$$S2 = \begin{bmatrix} fx2 & fy2 & \dots & Z2 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ fx20 & fy20 & \dots & Z20 \end{bmatrix} \quad (5)$$

Decision tree  $t$ ,

$$S2 = \begin{bmatrix} fx4 & fy4 & \dots & Z4 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ fx12 & fy12 & \dots & Z12 \end{bmatrix} \quad (6)$$

Where,  $S = S1 \cup S2 \cup \dots \cup St$

Equation 4, 5, and 6 shows that  $S1$  is the decision tree 1,  $S2$  is the decision tree 2,  $St$  is the decision tree  $t$  respectively and hence from the given dataset, the  $t$  decision trees are generated.

*Step 3:* The final decision values are evaluated from the decision trees and using random forest all the values are compared with the test sample classifier and hence the majority vote is determined. In the required class label the majority vote thus obtained is considered.

### D. Testing Process

Whether the decision is true or false is predicted initially by two types of values. If the set contain samples of different pattern, the entire data set is redefined into single pattern subsamples. Belonging to a single pattern, the decision tree is composed by a leaf when the set contains only samples. By purity measure of each node the feature selection is improved. As in equation 7 the general dataset can be converted into decision trees.

$$F(x) = F(x, a) \in F \rightarrow y = T(x) = \sum_{m=1}^{\hat{M}} c_m I \left[ x \in R_m \right] \quad (7)$$

Where,

Training set:  $\{y_i, x_{i1}, x_{i2}, \dots, x_{ik}\}_1^N = \{y_i, x_i\}$

K=#predictors and N=#samples

Score criterion to judge quality of fit of model,

$$R(a) = \frac{1}{N} \sum_{i=1}^N L(y_i, F(x_i, a)) \quad (8)$$

Squared error:

$$L(y, \hat{y}) = (y - \hat{y}) \quad (9)$$

Search strategy to minimize the score criterion:

$$\hat{a} = \arg \min_a \hat{R}(a) \quad (10)$$

$$\begin{aligned} & \left\{ \hat{C}_m, \hat{R}_m \right\}_1^M \\ & = \arg \min_{\left\{ \hat{C}_m, \hat{R}_m \right\}_1^M} \sum_{i=1}^N \left[ y_i - \sum_{m=1}^M C_m I(x_i \in R_m) \right]^2 \quad (11) \end{aligned}$$

There is more likely of over fitting training data as the tree grows larger. By pruning beyond the training data sub trees are generated. While pruning, the tree complexity (size) and goodness of data fit (node purity) are essential. By using the best splits in equation (8) & (9) pure children  $R^1$  and  $R^r$  are produced. Each tree is fully grown and unpruned based on the best split. Thus using equation (10) & (11) with various decision trees the

random forest model is generated. In accordance with the test sample the best class is predicted based on the majority voting.

1) *Estimating the test error:* While growing trees the test error is estimated. Out Of Bootstrap (OOB) [Breiman 2001] are samples that are not selected in bootstrap where almost 33-36% of samples for each tree grown. If they were novel test samples, predictions were made using OOB samples as input to the corresponding tree. For all OOB samples from all trees through book-keeping, the majority vote is computed after classification of dataset. In practice, with reasonable trees the estimated test error is very accurate. The final prediction is the mean class with maximum votes (classification). Using squared error loss by lowering prediction variance, while the bias remains unchanged bagging alone decreases test error.

#### IV. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, the evaluation result of proposed Random Forest Classifier are described and compared with existing LC-KNN and RC-KNN classification algorithm. The m value is important because it directly affects the real application and final performance. Thus, a group of experiments were conducted in order to select appropriate m for bringing great effect on five datasets by choosing different values of m. Specifically, on the datasets with m=10, 15, 20, 25 and 30, the LC-KNN, RC-KNN and Random Forest classifier in this group experiments were carried out respectively. Table 1 to 5 shows different values of m giving the comparison of classification performance and time cost of three algorithms (i.e., RC-KNN, LC-KNN, Random Forest classifier). Random forest algorithm is more interpretable and large number of predictors can be readily handled. During training for little additional computation the feature importance can be estimated. It generates an internal unbiased estimate of the test error as the forest building progresses.

Table 1. Classification accuracy (mean±standard deviation) and Time cost (seconds: mean±Standard deviation) on USPS dataset at different values of m

m value	Criteria	KNN		Our proposed algorithm
		LC-KNN	RC-KNN	
10	Accuracy	0.9027±1.6498e-005	0.9355±7.1306e-006	0.9487±7.2341e-006
	Time	3.5589±0.0107	3.7605±0.0242	3.8805±0.0257
15	Accuracy	0.8964±5.1803e-005	0.9338±4.1625e-006	0.9476±4.2385e-006
	Time	2.4857±0.0032	2.7260±0.0077	2.8954±0.0083
20	Accuracy	0.8770±7.4889e-005	0.9300±4.9238e-006	0.9445±4.967e-006
	Time	2.3202±0.0010	2.5157±0.01928	2.6007±0.01989
25	Accuracy	0.8793±4.9917e-005	0.9284±1.0637e-005	0.9421±1.0737e-005
	Time	1.8586±0.0008	1.9971±0.0042	1.9987±0.0054
30	Accuracy	0.8607±4.6629e-005	0.9275±1.1596e-005	0.9400±1.1654e-005
	Time	1.6441±0.0002	1.9249±0.0023	1.9311±0.0034

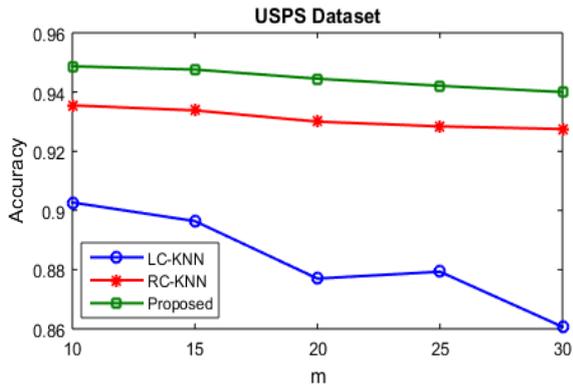


Fig.2. Classification accuracy of USPS dataset

The experimental results of LC-KNN and RC-KNN are compared with the results of our proposed system is shown in table 1. The accuracy and time cost of existing KNN algorithms LC-KNN and RC-KNN is shown in table 1 and using the USPS dataset it is compared to our proposed Random forest classifier. For handwritten digit recognition USPS (*United States Postal Service*) dataset is a standard dataset. Training is faster in Random forest

Table 2. Classification accuracy (mean±standard deviation) and Time cost (seconds: mean±Standard deviation) on MNIST dataset at different values of m.

	Criteria	KNN		Our proposed algorithm
		LC-KNN	RC-KNN	
10	Accuracy	0.7221±4.8878e-005	0.8389±3.1656e-005	0.9471±3.1689e-005
	Time	2.9369±0.0508	3.5504±0.0927	3.5654±0.0945
15	Accuracy	0.6840±2.3333e-004	0.8364±2.3136e-005	0.9332±2.3142e-005
	Time	2.8905±0.0456	3.1222±0.01397	3.1321±0.01632
20	Accuracy	0.6657±2.4739e-004	0.8353±3.3233e-005	0.9271±3.3421e-005
	Time	2.0564±0.0011	2.1490±0.0065	2.1511±0.0067
25	Accuracy	0.6478±2.2689e-004	0.8338±8.7844e-005	0.9176±8.7798e-005
	Time	1.8240±0.0020	2.1148±0.0094	2.1265±0.0097
30	Accuracy	0.6396±6.9156e-005	0.8313±3.8678e-005	0.9069±3.8698e-005
	Time	1.5457±0.0002	1.7274±0.0011	1.7326±0.0017

The above table enters the accuracy and time cost of existing KNN algorithms LC-KNN and RC-KNN and our proposed Random forest classifier using classification of MNIST dataset. In the field of machine learning, MNIST dataset is a massive database, which is used for training and testing. For training various image processing systems it includes handwritten digits that are used generally. The m values, the accuracy and time also increases and it is more efficient than the existing systems and also the complexity of big data is decreased which is seen from the table.

In accordance with the m values the MNIST ("Modified National Institute of Standards and Technology") dataset's classification accuracy level is described graphically. In accordance with the m values the graph compares the accuracy level of KNN and random forest. When compared to the existing KNN algorithm the accuracy level of random forest algorithm is increased. The accuracy level is decreased when the m values are increased. The entire dataset is clustered based on the generated decision trees. The classification is done where there is accuracy and less time consumption. Since

classifier algorithm which has fewer parameters also it plots the sample proximities and to visualize the output decision trees it is more interpretable. The m values are increased with decrease in time and accuracy is also altered and compared to the existing systems it is more efficient and thereby the complexity of big data is decreased to the greater extent which is shown in table below.

Fig. 2 shows the graphical representation of USPS dataset. In accordance with the m values the accuracy level of KNN and random forest is compared. When compared to the existing KNN algorithm the accuracy level of random forest algorithm is increased. From the graph it is clearly noted that when compared to the RC KNN, the LC KNN is very less accurate and in order to overcome limitations in data redundancy by RC KNN, the random forest classifier is used. The limitations of data redundancy and accuracy of medical big data are overcome by the Random Forest and hence in an accurate manner it can be dealt with big data with larger m values.

there is more accuracy this can be implemented in large datasets efficiently and less error is generated by larger volumes of datasets.

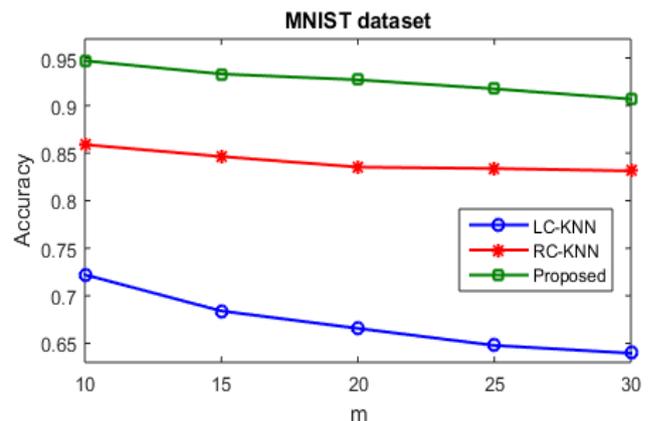


Fig.3. Classification accuracy of MNIST dataset

Table 3. Classification accuracy (mean±standard deviation) and Time cost (seconds: mean±standard deviation) on GISETTE dataset at different values of m.

m value	Criteria	KNN		Our proposed algorithm
		LC-KNN	RC-KNN	
10	Accuracy	0.9311±5.0989e-005	0.9526±1.4511e-005	0.9633±1.4576e-005
	Time	23.3933±0.9677	28.5940±3.2405	28.5978±3.2498
15	Accuracy	0.9252±1.0573e-004	0.9494±1.3378e-005	0.9611±1.3398e-005
	Time	18.0106±0.2434	23.1904±1.0894	23.1989±1.0898
20	Accuracy	0.9166±2.8267e-005	0.9411±5.4699e-004	0.9554±5.5111e-004
	Time	12.7685±0.0966	16.2759±0.8880	16.2801±0.8999
25	Accuracy	0.9150±7.0000e-005	0.9321±6.4810e-004	0.9500±6.4911e-004
	Time	9.9201±0.3696	13.8645±1.5093	13.8710±.5102
30	Accuracy	0.9079±1.0366e-004	0.9192±5.3796e-004	0.9411±5.4011e-004
	Time	8.4064±0.0784	11.3922±0.0658	11.3978±0.0698

The accuracy and time cost of existing KNN algorithms LC-KNN and RC-KNN is described in Table 3 and using the classification GISETTE dataset it is compared to our proposed Random forest classifier method. Used for training and testing in the field of machine learning GISETTE dataset is a handwritten digit recognition database. To separate the highly confusable digits 4 and 9 is the problem in this dataset. As the m values are increased, the time slightly decreases and accuracy is also altered after classification which is clear from the table but it is more efficient than the existing systems and hence decreases the complexity of big data.

In accordance with the m values, the accuracy level of KNN and random forest is compared which is described in the GISETTE dataset classification. In the m value, the accuracy is increased thus proving that classification accuracy increases well in medical big data which is shown in the graph. When compared to our proposed

methodology, the accuracy level of LC-KNN and RC-KNN is less.

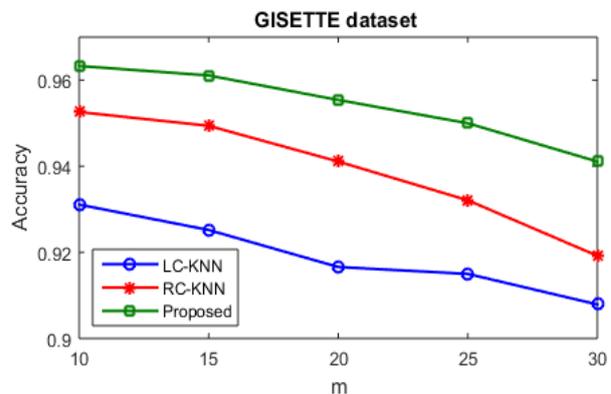


Fig.4. Classification accuracy of GISETTE dataset

Table 4. Classification accuracy (mean±standard deviation) and Time cost (seconds: mean±standard deviation) on LETTER dataset at different values of m.

m value	Criteria	KNN		Our proposed algorithm
		LC-KNN	RC-KNN	
10	Accuracy	0.7892±3.8822e-005	0.9495±1.0760e-006	0.9987±1.1454e-006
	Time	3.2391±0.0015	3.2994±0.0010	3.3104±0.0015
15	Accuracy	0.7932±3.7106e-005	0.9469±5.5751e-006	0.9900±5.5810e-006
	Time	3.3808±0.0435	3.4334±0.0585	3.4410±0.0590
20	Accuracy	0.6815±1.3812e-004	0.9451±1.9756e-006	0.9897±1.9811e-006
	Time	3.0938±5.8392e-004	3.1285±3.1243e-004	3.1340±3.1341e-004
25	Accuracy	0.7279±5.6480e-005	0.9423±5.2818e-006	0.9885±5.2901e-006
	Time	3.3950±0.0018	3.4813±0.0054	3.4902±0.0067
30	Accuracy	0.6214±9.8480e-005	0.9403±3.9204e-006	0.9872±3.9342e-006
	Time	3.0889±1.3000e-003	3.1168±3.8514e-004	3.1453±3.8621e-004

The accuracy and time of existing KNN algorithms LC-KNN and RC-KNN is shown in table 4 and using LETTER dataset classification it is compared to our proposed Random forest classifier. A Multi-class classification dataset is the LETTER dataset. As the m values increases accuracy, time is increased accordingly which are obvious from the table but compared to the already existing systems it is more efficient and hence the complexity of big data decreases.

The accuracy level of KNN and random forest in accordance with the m values is compared by the graphical representation of LETTER dataset classification.

When compared to the existing KNN algorithm, the accuracy level of random forest algorithm is increased. Our proposed algorithm works well in classification accuracy for medical big data as the m value increases the accuracy is increased.

The accuracy and time of existing KNN algorithms LC-KNN and RC-KNN is shown in table 5 and using PENDIGITS dataset classification, it is compared to our proposed Random forest classifier. A Multi-class classification dataset is the PENDIGITS dataset. The m values increases with increase in accuracy and time accordingly which is clear from the table but compared to

the existing systems it is more efficient and the complexity of big data is decreased. In accordance with the m values, the graphical representation of PENDIGITS dataset classification compares the accuracy level of KNN and random forest. When compared to the existing

KNN algorithm the accuracy level of random forest algorithm is increased. In existing systems, when the m values are increased the accuracy level is decreased but increases the accuracy of random forest algorithm.

Table 5. Classification accuracy (mean±standard deviation) and Time cost (seconds: mean±standard deviation) on PENDIGITS dataset at different values of m.

m value	Criteria	KNN		Our proposed algorithm
		LC-KNN	RC-KNN	
10	Accuracy	0.9452±3.5382e-005	0.9721±4.7991e-006	0.9943±4.8122e-006
	Time	2.3380±0.0041	2.4056±0.0101	2.4311±0.0145
15	Accuracy	0.9316±1.0341e-004	0.9711±6.0196e-006	0.9923±6.0210e-006
	Time	2.5451±0.0011	2.5709±0.0089	2.5899±0.0090
20	Accuracy	0.9163±1.5515e-004	0.9700±2.5390e-006	0.9845±2.5541e-006
	Time	2.2233±6.4795e-005	2.2554±2.1569e-004	2.2670±2.1670e-004
25	Accuracy	0.9216±1.5677e-004	0.9687±3.5642e-006	0.9815±3.5734e-006
	Time	2.5270±0.0056	2.5468±0.0083	2.5623±0.0091
30	Accuracy	0.9088±1.8409e-004	0.9683±1.5809e-006	0.9810±1.5967e-006
	Time	2.1805±7.4785e-005	2.2022±8.9611e-005	2.2632±8.9678e-005

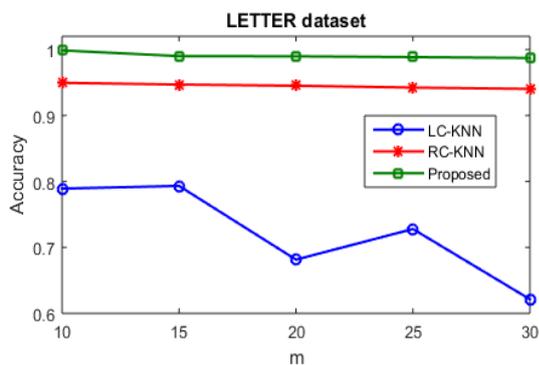


Fig.5. Classification accuracy of LETTER dataset

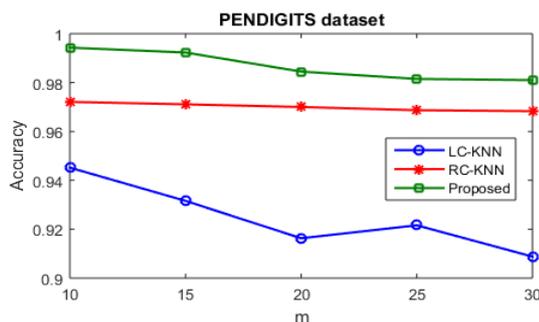


Fig.6. Classification accuracy of PENDIGITS dataset

#### A. Performance comparison of Random Forest Classifier & KNN

When compared to the existing systems LC KNN and RC KNN the proposed algorithm Random forest classifier works well in terms of accuracy and time cost for classification of big data which is shown from the experimental results on the classification of five datasets namely USPS, MNIST, GISETTE, LETTER, PENDIGITS. The accuracy level for individual datasets is demonstrated in the graphical representation. When the m value increases the time increases and also the accuracy is increased in big data which is described in table 6. The major advantage of our proposed system is

that the accuracy level increases when the value of m increases. Thus in big data classification our system can be used efficiently. Hence in terms of classification of accuracy and time our proposed system works well.

Table 6. Classification Accuracy and Time Cost of KNN algorithm and Random forest algorithm on five dataset

Dataset	KNN		Our Proposed Algorithm	
	Accuracy	Time	Accuracy	Time
USPS	0.9482	32.8764	0.9611	33.1007
MNIST	0.8635	24.1575	0.8890	24.9874
GISETTE	0.9660	217.3327	0.9983	220.0110
LETTER	0.9518	19.8246	0.9518	20.0178
PENDIGITS	0.9780	7.2982	0.9945	7.6530

## V. CONCLUSION

In this work initially, the k-means algorithm is used for clustering the data and then by using certain attributes clustered datasets are randomly selected and decision trees are generated from the clustered data. The decision values are taken into account from the decision trees. Using random forest classifier the decision values are classified in accordance with test samples thereby creating the required class label from the majority votes produced. Using Random forest classifier the medical big data can be classified well compared to the existing LC-KNN and RC-KNN which is shown by the experimental results. Large number of predictors can readily be handled by Random forest algorithm and also it is more interpretable. It generates an internal unbiased estimate of the test error in the forest building progresses. Hence the accuracy is increased by our proposed work and hence capable of dealing with classification of medical big data.

## REFERENCES

- [1] U. Sivarajah, M. Kamal, Z. Irani and V. Weerakkody, "Critical analysis of Big Data challenges and analytical methods", *Journal of Business Research*, vol. 70, no.1, pp.

- 263-286, Jan 2017.
- [2] A. Azar and A. Hassanien, "Dimensionality reduction of medical big data using neural-fuzzy classifier", *Soft Computing*, vol. 19, no. 4, pp. 1115-1127, June 2014.
- [3] A. Nega and A. Kumlachew, "Data Mining Based Hybrid Intelligent System for Medical Application", *International Journal of Information Engineering and Electronic Business*, vol. 9, no. 4, pp. 38-46, 2017.
- [4] I. Hashem, I. Yaqoob, N. Anuar, S. Mokhtar, A. Gani and S. Ullah Khan, "The rise of "big data" on cloud computing: Review and open research issues", *Information Systems*, vol. 47, No.1, pp. 98-115, January 2015.
- [5] C. Philip Chen and C. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data", *Information Sciences*, vol. 275, No.22, pp. 314-347, August 2014.
- [6] F. Costa, "Big data in biomedicine", *Drug Discovery Today*, vol. 19, no. 4, pp. 433-440, 2014.
- [7] F. Shen, Q. Ouyang, W. Kasai and O. Hasegawa, "A general associative memory based on self-organizing incremental neural network", *Neurocomputing*, vol. 104, no.6, pp. 57-71, March 2013.
- [8] A. Altaher, "An improved Android malware detection scheme based on an evolving hybrid neuro-fuzzy classifier (EHNFC) and permission-based features", *Neural Computing and Applications*, vol.27, no. 1, pp.1-11, November 2016.
- [9] D. Dimitrov, "Medical Internet of Things and Big Data in Healthcare", *Healthcare Informatics Research*, vol. 22, no. 3, p. 156, July 2016.
- [10] K. Panihar and V. Verma, "A Study Some Data Mining Classification Techniques", *International Journal of Modern Trends in Engineering & Research*, vol. 4, no. 1, pp. 210-215, Jan 2017.
- [11] N. Das, L. Das, S. Swarup Rautaray and M. Pandey, "Big Data Analytics for Medical Applications", *International Journal of Modern Education and Computer Science*, vol. 10, no. 2, pp. 35-42, 2018.
- [12] G. M S, N. R and S. Prabhu, "High Performance Computation of Big Data: Performance Optimization Approach towards a Parallel Frequent Item Set Mining Algorithm for Transaction Data based on Hadoop MapReduce Framework", *International Journal of Intelligent Systems and Applications*, vol.9, no. 1, pp.75, 2017.
- [13] J. Zhang and J. Yang, "Linear reconstruction measure steered nearest neighbor classification framework", *Pattern Recognition*, vol. 47, no. 4, pp. 1709-1720, April 2014.
- [14] Y. Chen, J. Yang, S. Liou, G. Lee and J. Wang, "Online classifier construction algorithm for human activity detection using a tri-axial accelerometer", *Applied Mathematics and Computation*, vol. 205, no. 2, pp. 849-860, November 2008.
- [15] E. Althagafy and M. Jameel Qureshi, "Novel Cloud Architecture to Decrease Problems Related to Big Data", *International Journal of Computer Network and Information Security*, vol. 9, no. 2, pp. 53-60, Feb 2017.
- [16] Z. Benmounah, S. Meshoul and M. Batouche, "Scalable Differential Evolutionary Clustering Algorithm for Big Data Using Map-Reduce Paradigm", *International Journal of Applied Metaheuristic Computing*, vol. 8, no.1, pp. 45-60, Jan 2017.
- [17] G. Luo, "PredicT-ML: a tool for automating machine learning model building with big clinical data", *Health Information Science and Systems*, vol. 4, no. 1, 2016.
- [18] M. Pratama, J. Lu and G. Zhang, "Evolving Type-2 Fuzzy Classifier", *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 3, pp. 574-589, June 2016.
- [19] P. Gutiérrez, M. Lastra, J. Bacardit, J. Ben fez and F. Herrera, "GPU-SME-kNN: Scalable and memory efficient kNN and lazy learning using GPUs ", *Information Sciences*, vol. 373, no.9, pp. 165-182, December 2016.
- [20] Y. Zhang, J. Ren, J. Liu, C. Xu, H. Guo and Y. Liu, "A Survey on Emerging Computing Paradigms for Big Data", *Chinese Journal of Electronics*, vol. 26, no. 1, pp. 1-12, Jan 2017.
- [21] Y. Wang and N. Hajli, "Exploring the path to big data analytics success in healthcare", *Journal of Business Research*, vol. 70, no.1, pp. 287-299, Jan 2017.
- [22] P. Marcheschi, "Relevance of eHealth standards for big data interoperability in radiology and beyond", *La radiologia medica*, vol. 122, no. 6, pp. 437-443, November 2016.
- [23] D. Dimitrov, "Medical Internet of Things and Big Data in Healthcare", *Healthcare Informatics Research*, vol. 22, no. 3, p. 156, July 2016.

#### Authors Profiles



#### **R.Saravana kumar ( Ramachandran)**

obtained B.E Degree in computer science and Engineering from Bharathiyar University, Coimbatore, tamilnadu, india in 2003. He obtained M.E degree in computer sciences and Engineering from Anna University, Chennai in 2007 .He has done PhD in Data Science from Anna University, Chennai in the year of 2015. Currently he is working as a professor in computer science and Engineering department from Dayananda Sagar Academy of Technology and Management, Bangalore, India. His area of Interest is Data Science.



#### **P.Manikandan (Parasuraman Manikandan)**

obtained his B.E Degree in Computer Science and Engineering from Bharathiyar University, Coimbatore, Tamilnadu, India in 1996. Then obtained M.E Degree in Computer Science and Engineering from Anna University, Chennai, Tamilnadu in 2008 and Ph.D degree from the Anna University Chennai during 2009-2016. Currently, He is working as a Professor in Computer Science and Engineering Department from Malla Reddy Engineering College for Women, Maisammaguda, Secunderabad, Telangana, India. His research interest is in Data Mining.

**How to cite this paper:** R. Saravana kumar, P. Manikandan, "Medical Big Data Classification Using a Combination of Random Forest Classifier and K-Means Clustering", *International Journal of Intelligent Systems and Applications(IJISA)*, Vol.10, No.11, pp.11-19, 2018. DOI: 10.5815/ijisa.2018.11.02