Research Journal of Applied Sciences, Engineering and Technology 7(17): 3633-3642, 2014
DOI:10.19026/rjaset.7.716
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: November 21, 2013 Accepted: December 18, 2013 Published: May 05, 2014

Research Article

A Novel Ensemble Classifier based Classification on Large Datasets with Hybrid Feature Selection Approach

[1]J. Vandar Kuzhali and [2]S. Vengataasalam
[1]Department of Computer Applications, Erode Sengunthar Engineering College, India
[2]Department of Mathematics, Kongu Engineering College, India

Abstract: Exploring and analyzing large datasets has become an active research area in the field of data mining in the last two decades. There had been several approaches available in the literature to investigate the large datasets that comprise of millions of data. The most important data mining approaches involved in this task are preprocessing, feature selection and classification. All the three approaches have their own importance in carrying out the task effectively. Most of the existing techniques suffer from drawbacks of high complexity and computationally costly on large data sets. Especially, the classification techniques do not provide consistent and reliable results for large datasets which makes the existing classification systems inefficient and unreliable. This study mainly focuses on develop a novel and efficient framework for analyzing and classifying a large dataset. This study proposes a novel classification approach on large datasets through the process of ensemble classification. Initially, efficient preprocessing approach based on enhanced KNN and feature selection based on genetic algorithm integrated with Kernal PCA are carried out which selects a subset of informative attributes or variables to construct models relating data. Then, Classification is carried on the selected features based on the ensemble approach to get accurate results. This research study presents two types of ensemble classifiers called homogenous and heterogeneous ensemble classifiers to evaluate the performance of the proposed system. Experimental results shows that the proposed approach provide significant results for various large datasets.

Keywords: ANFIS, classification, datasets, dimensionality reduction, enhanced KNN, ensemble classifier, feature selection, FRB, fuzzy classifier, preprocessing

INTRODUCTION

Classification is the process which is widely used in human activity. The main goal of the classification of the data is to arrange and classify the data in distinct classes (Purnami *et al*., 2011). At present, numerous applications utilize very large data sets of high dimensionality, therefore classifying, understanding this information becomes a very hard task. Data and web mining, text categorization, financial forecasting and biometrics are some areas in which enormous amounts of information have to be employed.

The processing of a very large data set suffers from a major difficulty such as high storage and time requirements. It is observed that, a large data set cannot be fully stored in the internal memory of a computer. On the other hand, the time needed to learn from such a whole data set can become prohibitive. These issues become worse in the case of using distance-based learning algorithms, such as the Nearest Neighbor rule (Cover and Hart, 1967; Dasarathy, 1990), due to its apparent necessity of indiscriminately storing all the training instances.

Moreover, the occurrence of noise and unrelated features in large data sets make the analysis process more complicated. For example, microarray data generally contains thousand of features genes with only few dozen of samples or patterns (Somorjai *et al*., 2003). Evidently, selecting a relevant number of subset of features in large high dimensional data sets is a difficult process. A number of preprocessing and feature selection algorithms had been available in the literature for dimensionality reduction of the large datasets. But, most of the feature selection approaches do not provide consistent results and some of the relevant features are supposed to be missed (Stefano *et al*., 2008).

Hence, this approach uses a hybrid fusion of feature selection approaches (Yang *et al*., 2010). Neural network based approaches are observed to provide significant and consistent results in pattern recognition and classification. Hybrid neural network approaches which are the fusion of two approaches are the recent promising trend in data mining. So in this research study, an Enhanced KNN (Dhaliwal *et al*., 2011) method is used to find the missing values from the

Corresponding Author: J. Vandar Kuzhali, Department of Computer Applications, Erode Sengunthar Engineering College, India

whole dataset by preprocessing process. Feature selection of the datasets is done using Enhanced Genetic Algorithm combined with Kernel PCA_SVM Algorithm (Fangjun *et al*., 2012).

Classification is the essential and critical section that has to be carefully formulated for the process of analyzing the large dataset. Although, some amount of the features can lead to high classification accuracy, the extra features added over there cannot contribute much to the performance but they do not humiliate the general performance. Then, the classifier is expected to classify unlabeled instances into one or more predefined categories based on their content. The applications that comprise of millions of attributes or variables would make the pattern classification or prediction difficult which in turn results in inefficient classification with lesser accuracy (Khan *et al*., 2001).

An efficient and promising choice to process large data set is to independently learn from a number of moderate-sized subsets and integrate their results through ensemble of classifiers. Ensemble classification is one of the most recent approaches widely used in pattern recognition and machine learning. It is a potential approach which basically comprise of integrating the results from multiple classifiers. The main goal of the ensemble is to attain significant classification accuracy than that offered by its individual classifiers with a lesser complexity (Shipp and Kuncheva, 2002).

Ensemble classification can be categorized into homogenous and heterogeneous classification techniques. Homogenous approaches comprises of only one classifier for ensemble. But, ensemble heterogeneous classification comprises of different classifiers for ensemble.

Hence, in this research study, an efficient classification process is carried out using the both homogeneous and heterogeneous ensemble classifiers using fuzzy based classifiers.

METHODOLOGY

The proposed methodology is discussed as follows is shown in Fig. 1.

In general level, the data sets can be characterized by their size and the type. Size can be classically measured by the Number (N) of individual objects or patterns contained in the data set and the dimensionality (d) of each individual object that is, the number of measurements, variables, features, or attributes recorded for each object (Guha *et al*., 1998).

The main objective is to store data sets that are large enough to endow with challenges to existing algorithms in terms of scaling behavior as a function of N and d, so far that are not so large as to make downloading through the Internet in reasonable time is not possible. Therefore, the target individual data sets up to 1000 Megabytes in size, which approximately permit for the storage of an N = 500; 000 measurements ×d = 100 dimensional data set with 8 bytes per measurement and no compression.

Preprocessing using enhanced KNN imputation: In preprocessing process, the missing values of the datasets from the whole dataset are regained. Missing data imputation is a process that changes the missing values with various feasible values. Imputed values are indulgence as dependable as the truly observed data, but they are simply as fine as the assumption used to build them. Outliers are the noisy data which do not converse to the inherent model that created the data under surveillance. From Hart (1967) outliers are notice that should be eradicated so as to improve the accuracy of clustering process.

Feature selection using KPCA SVM with GA model: Kernel principal component analysis: Principal Component Analysis (PCA) is an ordinary method used for the purpose of dimensionality reduction and feature extraction (Bin *et al*., 2009). It can remove only the linear structural data in the data set but cannot remove this nonlinear structure information. Kernel Principal Component Analysis (KPCA) is an advanced technique than PCA, which extracts principal components by accepting a nonlinear kernel method (Liao and Jiang, 2008; Ding *et al*., 2009). A key approaching behind KPCA is to transform the input data into a high dimensional feature space F in which PCA is carried out and in execution, the implicit feature vector in F does not need to be calculated openly, at the same time as it is just ended up by calculating the inner product of two vectors in F with a kernel function.

KPCA SVM model: After the process of feature extraction by KPCA, the training data points can be denoted as (t1, y1), (t2; y2), …., (tp; yp), $t_i \in R^n (n < m)$ is the transformed input vector, $y_n \in R^n$ is

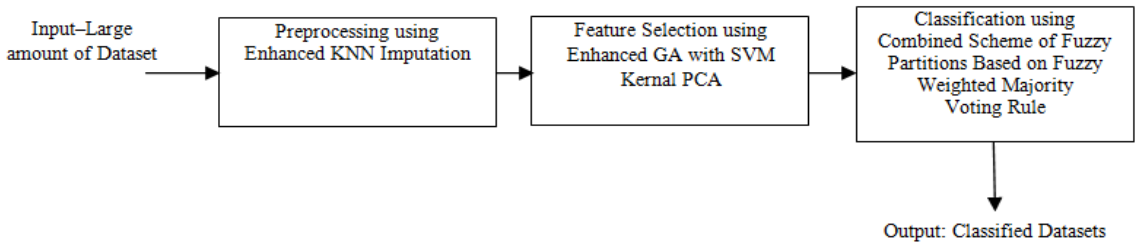

Fig. 1: Proposed methodology

the target value. The brief explanation of the process is studied in Dhaliwal *et al*. (2011).

Enhanced GA for parameter selection of KPCA SVM model: In this study, the choice of the three positive parameters, σ, ε and C of KPCA SVM representation is significant to the accuracy of the classification for large datasets. Hence, enhanced genetic algorithms are combined with the proposed KPCA SVM model to optimize the parameter selection. A negative Mean Absolute Percentage Error (MAPE) is used as the fitness function for calculating the fitness value (Pai and Hong, 2005). The MAPE is represented as follows:

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{a_i - f_i}{a_i}\right| \times 100\% \quad (1)$$

where, a_i and f_i represent the actual and forecast values and N is the number of classification forecasting periods. The enhanced GA is used to capitulate smaller MAPE by searching for enhanced combinations of three parameters in KPCA SVM, which is explaining below:

Step 1: The formation of an initial population of chromosomes is done. The three free parameters σ, ε and C are programmed in a binary format and are represented by a chromosome (Fangjun *et al*., 2012).

Step 2: The fitness function value of each chromosome is calculated by the cross-validated projecting accuracy of the SVM model. Based on fitness functions, chromosomes with higher fitness values are further likely to give up offspring in the next generation. The roulette wheel selection principle is applied to select chromosomes for reproduction.

Step 3: **Crossover and mutation:** Mutations are processed arbitrarily by changing a '1' bit into a '0' bit or a '0' bit in to a '1' bit. The single-point crossover principle is in use. Segments of paired chromosomes among two single-minded break-points are interchanged. The rates of crossover and mutation are probabilistically found out. In this investigation, the probabilities of crossover and mutation are set to 0.5 and 0.1, respectively.

Step 4: A new population is created for the next generation.

Step 5: If the number of generations equals a given scale, then stop; else go to step 2.

Step 6: Obtain the optimal parameters σ, ε and C of the KPCA SVM model (Fangjun *et al*., 2012).

Hence, the optimal features are selected and the selected features are used for final classification purpose is seen below.

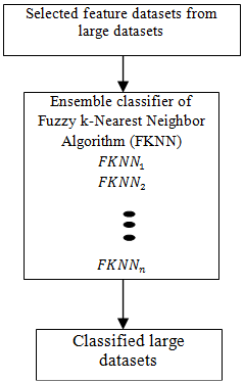

Fig. 2: Fuzzy k-nearest neighbor classifier

Proposed ensemble based classification approach: The main aim of ensemble methodology is to construct a predictive framework by combining multiple models. This ensemble framework can be used for improving prediction accuracy and the overall system performance. In recent years, ensemble classification has been widely used in various disciplines of science and engineering. The fundamental notion of ensemble methodology is to weigh several individual classifiers and then integrate them in a single classifier that outperforms every one of them (Lior, 2010). This research study uses both homogeneous and heterogeneous ensemble classifiers in order to improve the overall results of the classification process. Then, the performance analyses between homogeneous and heterogeneous classifiers are carried out to determine the best classifier for the chosen task.

Homogeneous ensemble classifier (Fuzzy K-Nearest Neighbor algorithm (FKNN)): The optimal features selected by the feature selection process undergone classification using homogeneous ensemble classifier. The classifier used in this approach is Fuzzy k-Nearest Neighbor classifier which gives better classification when compared with other traditional ensemble classifiers. Initially, the selected feature datasets are given to the classifier, in which the datasets are classified in each layer and gives better results. The process is shown below.

The K-Nearest Neighbor algorithm (KNN) shown in Fig. 2 is a non parametric pattern classification method (Hojjatoleslami and Kittler, 1996) used widely in the field of classification. In 1985, a fuzzy based KNN by building the fuzzy set assumption into the KNN algorithm is called as Fuzzy KNN classifier algorithm" (FKNN) (Keller, 1985). Unlike the individual KNN classes, in this approach, the fuzzy memberships of samples are allocated to various groups by the following procedure:

$$u_i(x) = \frac{\sum_{j=1}^{k} u_{ij}\left(\frac{1}{\left\|x - x_j\right\|^{\frac{2}{(m-1)}}}\right)}{\sum_{j=1}^{k}\left(\frac{1}{\left\|x - x_j\right\|^{\frac{2}{(m-1)}}}\right)} \quad (2)$$

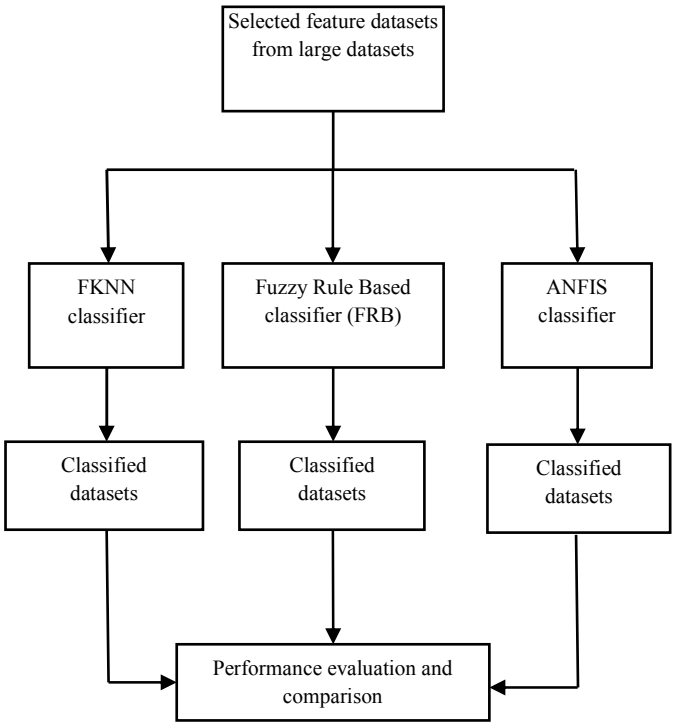

Fig. 3: Heterogeneous ensemble classifier

where, $i = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, k$, where c represents number of classes and k denotes the number of nearest neighbors. The fuzzy parameter denoted by 'm' is used to choose how intensely the distance is weighted when computing each neighbor's influence to the membership value and its value is normally chosen as $m \in (1; +\infty)$ (Chen *et al*., 2011a). $\|x - x_j\|$ is the Euclidean distance between x and its jth nearest neighbor x_j. And u_{ij} is the membership degree of the pattern x_j from the training set to the class i, among the k nearest neighbors of x. u_{ij} can be modeled in two forms namely the crisp membership form in which each training pattern has whole membership in their known class and non-memberships in all other classes (Chen *et al*., 2011b). The second form is the constrained fuzzy membership in which the k nearest neighbors of each training pattern namely (x_k) are identified and the membership of x_k in each class is allocated as:

$$u_{ij}(x_k) = \begin{cases} 0.51 + \left(\frac{n_j}{K}\right) * 0.49, if\ j = i \\ \left(\frac{n_j}{K}\right) * 0.49, otherwise \end{cases} \quad (3)$$

The value n_j represents the number of neighbors identified which equivalent to the j^{th} class. It is observed that the second way gives better results in terms of its accuracy. After computing all the memberships for a query sample, it is allotted to the class with the highest membership value.

Heterogeneous ensemble classifier: The optimal features selected by the feature selection process undergone classification using heterogeneous ensemble classifier. The classifiers used in this approach are Fuzzy k-Nearest Neighbor classifier, ANFIS Classifier and FRB classifier which give better classification results when compared with other conventional ensemble classifiers. Initially, the selected feature datasets are given to the classifier, in which the datasets are classified in each layered of the each classifier and gives better results. The process of the classification is shown below in Fig. 3.

Adaptive Neuro-Fuzzy Inference System (ANFIS) classifier: The intention of classification system is to allocate each input to one of 'c' pattern classes. It is the method of assigning a label to each anonymous input data. A neuro fuzzy approach called ANFIS used to classify the large datasets. The performance measures used in this study are classification accuracy and convergence rate. The results are compared with the neural classifier and the fuzzy classifier to show the better nature of ANFIS (Jang, 1993).

Architecture of ANFIS: The ANFIS is a fuzzy Sugeno model lay in the structure of adaptive systems to make simple learning and adaptation (Jang, 1993). Such structure gives the ANFIS modeling more resourceful and less dependent on proficient knowledge. The ANFIS structural design is presented by two fuzzy if-then rules based on a first order Sugeno model are calculated as:

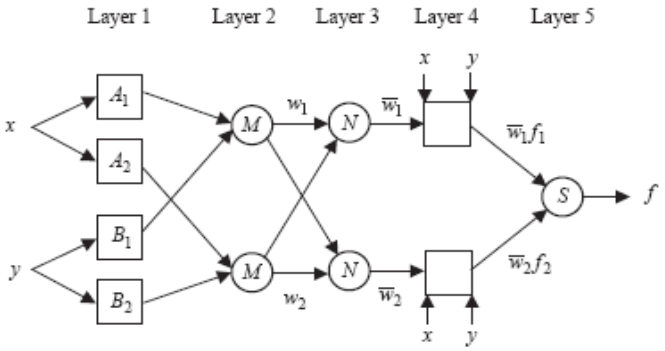

Fig. 4: ANFIS architecture

$$Rule\ 1: If\ (x\ is\ A1) and\ (y\ is\ B1) then\ (f1 = p1x + q1y + r1) \quad (4)$$

$$Rule\ 2: If\ (x\ is\ A2) and\ (y\ is\ B2) then\ (f2 = p2x + q2y + r2)$$

where, x and y are the inputs of the ANFIS model.

A_i and B_i are the fuzzy sets, f_i are the outputs through the fuzzy part defined by the fuzzy rule, pi; qi and ri are the design parameters predicted during the training process. Figure 4 shows the ANFIS architecture with in the form of two rules in which a circle point out a fixed node, while a square indicates an adaptive node.

The nodes in the first layer are the adaptive nodes and the outputs produced are the fuzzy membership grade which are given by:

$$O_i^1 = \mu_{Ai}(x), i = 1,2 \quad (5)$$

$$O_i^1 = \mu_{Bi-2}(y), i = 3,4 \quad (6)$$

where, $\mu_{Ai}(x)$, $\mu_{Bi-2}(y)$ can adopt any fuzzy membership function. For instance, if the bell shaped membership function is used, $\mu_{Ai}(x)$ is given by:

$$\mu_{Ai}(x) = \frac{1}{1+\{\left(\frac{x-c_i}{a_i}\right)^2\}^{b_i}} \quad (7)$$

where, a_i, b_i and c_i represent the parameters of the membership function, managing the bell-shaped functions.

The ANFIS Architecture is shown in the Fig. 4.

The nodes are fixed a node which is to be presented in a second layer. They are labeled with M, representing that they carry out as an easy multiplier. The outputs of this layer can be correspond to as:

$$O_i^2 = w_i = \mu_{Ai}(x)\mu_{Bi}(y), i = 1,2 \quad (8)$$

which are the called as firing strengths of the rules.

The nodes in the third layer are also fixed nodes and are denoted with N, representing their normalization position to the firing strengths from the preceding layer.

The outputs of this layer can be correspond to as:

$$O_i^3 = \overline{w}_i = \frac{w_i}{w_1+w_2}, i = 1,2 \quad (9)$$

which are the so-called normalized ring strengths.

The nodes in the fourth layer are considered as the adaptive nodes. Each node forms an output based on the product of the normalized firing strength and a first-order polynomial. Thus, the outputs of this layer are given by:

$$O_i^4 = \overline{w}_i f_i = \overline{w}_i(p_i x + q_i y + r_i), i = 1,2 \quad (10)$$

There is a single fixed node in the fifth layer indicated with S. This node carries out summation of all incoming signals. Therefore, the overall output of the framework is given by:

$$O_i^5 = \sum_{i=1}^2 \overline{w}_i f_i = \frac{(\sum_{i=1}^2 w_i f_i)}{w_1+w_2}, i = 1,2 \quad (11)$$

It can be experiential that there are two adaptive layers in this ANFIS structural design, that is the first layer and the fourth layer. In the first layer, there are three changeable parameters $\{a_i, b_i, c_i\}$ which are connected to the input membership functions are called as basis parameters. In the fourth layer, there are also three adjustable parameters $\{p_i, q_i, r_i\}$, pertaining to the first order polynomial. These parameters are so-called consequent parameters (Jang, 1993).

Learning algorithm of ANFIS: The main aim of the learning algorithm for this architecture is to alter all the adjustable parameters, namely $\{a_i, b_i, c_i\}$ and $\{p_i, q_i, r_i\}$, to make the ANFIS output match the training data (Jang, 1992). When the basis parameters a_i, b_i and c_i of the membership function are fixed, the output of the ANFIS model can be defined as:

$$f = \frac{w_1}{w_1+w_2} f_1 + \frac{w_2}{w_1+w_2} f_2 \quad (12)$$

Substituting Eq. (8) into (11) yields:

$$f = \bar{w}_1 f_1 + \bar{w}_2 f_2 \quad (13)$$

By substituting the fuzzy if-then rules into Eq. (12), it becomes:

$$f = \bar{w}_1(p_1 x + q_1 y + r_1) + \bar{w}_2(p_2 x + q_2 y + r_2) \quad (14)$$

After rearrangement, the output can be expressed as:

$$f = (\bar{w}_1 x)p_1 + (\bar{w}_1 y)q_1 + (\bar{w}_1)r_1 + (\bar{w}_2 x)p_2 + (\bar{w}_2 y)q_2 + (\bar{w}_2)r_2 \quad (15)$$

which is a linear combination of the variable resultant parameters p_1, q_1, r_1, p_2, q_2 and r_2. The least squares approach can be utilized to categorize the optimal values of these parameters. Then, the maximum repeated pixel intensity of the large datasets is determined. In order to determine the maximum repeated pixel, initially, the intensities of all the pixels of the large datasets have to be identified through histogram and then all the pixels of dataset are compared with each other. After determining the maximum repeated dataset, the result is given to the classifier. Similarly, the maximum repeated data in the whole dataset is determined and its result is given to the classifier. The classifier classifies the large datasets by comparing all the features of the dataset (Jang, 1992).

Fuzzy Rule-Based classifier (FRB): A large number of approaches are available in the literature to carry out the classification task. Amongst them, FRBCSs provide an interpretable replica through linguistic labels in their rules (Sanz *et al*., 2011):

Consider m labeled patterns $x_p = (x_{p1}, \ldots\ldots, x_{pn})$, $p = 1, 2, \ldots, m$.

where, x_{pi} is the i^{th} attribute value ($i = 1, 2, \ldots, n$). A set of linguistic values and their membership functions are available to describe each and every attribute. The fuzzy rules are used based on the following form:

Rule R_j: If x_1 is A_{j1} and .... and x_n is A_{jn} then Class = C_j with RW_j (16)

where, R_j denotes the label of the jth rule, $x = (x1, \ldots, xn)$ is an n-dimensional pattern vector, A_{ji} represents an antecedent fuzzy set on behalf of a linguistic term, C_j denotes a class label and RW_j represents the rule weight. Specially, the rule weight is evaluated through the Penalized Certainty Factor as:

$$PCF_j = \frac{\sum_{x_p \in Class C_j} \mu A_j(x_p) - \sum_{x_p \notin Class C_j} \mu A_j(x_p)}{\sum_{p=1}^{m} \mu_{Aj}(x_p)} \quad (17)$$

Let $x_p = (x_{p1}, \ldots\ldots, x_{pn})$ be a new pattern, L denotes the number of rules in the rule base and M represents the number of classes of the problem. The steps of the FRM are as follows (Sanz *et al*., 2011).

Matching degree: This matching degree facilitates the activation of the if-part for all rules in the rule base with the pattern x_p. A conjunction operator (t-norm) is functional to perform this computation:

$$\mu_{Aj}(x_p) = T(\mu A_{j1}(x_{p1}), \ldots\ldots, \mu A_{jn}(x_{pn})), j = 1, \ldots\ldots L \quad (18)$$

Association degree: This degree helps in the evaluation of the association degree of the pattern x_p with the M classes based on each rule in the rule base. When the rules shown in Eq. (16) is used, this association degree only refers to the consequent class of the rule (i.e., k = Class (R_j)):

$$b_j^k = h\left(\mu_{Aj}(x_p), RW_j^k\right), k = 1, \ldots\ldots, M, j = 1, \ldots\ldots l \quad (19)$$

Pattern classification soundness degree for all classes: An aggregation function that integrates the positive degrees of association calculated from the association degree step is utilized:

$$Y_k = f\left(b_j^k, j = 1, \ldots\ldots, L \text{ and } b_j^k > 0\right), k = 1, \ldots\ldots M \quad (20)$$

Classification: A decision function F is utilized over the soundness degree of the model for the pattern classification for all classes. This formulation identifies the class label l based on the maximum value:

$$F(Y_1, \ldots . Y_M) = \arg\max(Y_k) . k = 1, \ldots . M \quad (21)$$

The classifier classifies the large datasets by comparing all the features of the dataset.

EXPERIMENTAL RESULTS

To evaluate the experiment, the experiments are carried out using UCI benchmark data. Initially, the preprocessing process is carried out then the feature selection algorithm is done and the features are selected, the results are shown below.

In several machine learning algorithms, there are two structures of high-dimensional data. By tradition, the dimensionality is generally considered to be high if data may contain hundreds of features. In that form of data, the number of occurrences is generally much larger than the dimensionality. In the novel fields such as text classification and genomic microarray study, the dimensionality is in the order of thousands and

Table 1: Datasets from UCI benchmark data

Title	Features	Instances	Classes
Lung cancer	56	32	3
Promoters	57	106	2
Splice	60	3190	3

Table 2: Preprocessing process results

Title	KNN	Enhanced KNN
Lung cancer	22	12
Promoters	28	16
Splice	31	23

Table 3: Results of feature selection process

Title	GA_KPCA	GA_KPCA SVM	Enhanced GA_KPCA SVM
Lung cancer	11	8	3
Promoters	9	4	5
Splice	21	16	10

Table 4: Comparative results of feature selection accuracy

Techniques	Accuracy in (%)
GA_KPCA	79.5
GA_KPCA SVM	85.3
Enhanced GA_KPCA SVM	91.2

regularly greatly exceeds the number of instances. Therefore, the proposed method is evaluated in comparison with others on high-dimensional data of both forms.

UCI benchmark data: All together 10 data sets in the traditional form are selected from the UCI Machine Learning Repository and the UCI KDD Archive.2 These data sets contain various numbers of features, instances and classes, as shown in Table 1. For each data set, first run the entire preprocessing and feature selection algorithm in comparison and obtain the running time and selected features for each algorithm.

Results on preprocessing process: Table 2 shows the comparative results of KNN and Enhanced KNN during preprocessing process.

Figure 5 shows the comparative values of KNN and Enhanced KNN during preprocessing process. From the above graph, it is to be noted that the proposed enhanced KNN gives better results than KNN.

On feature selection process: Table 3 shows the comparison results of feature selection process of various techniques.

Figure 6 shows the comparison results of feature selection process of various techniques. It is to be noted that, the proposed Enhanced GA_KPCA SVM approach performs better than the other existing GA_KPCA and GA_KPCA SVM approaches.

Feature selection accuracy: Table 4 shows the comparative analysis of feature selection accuracy of various approaches.

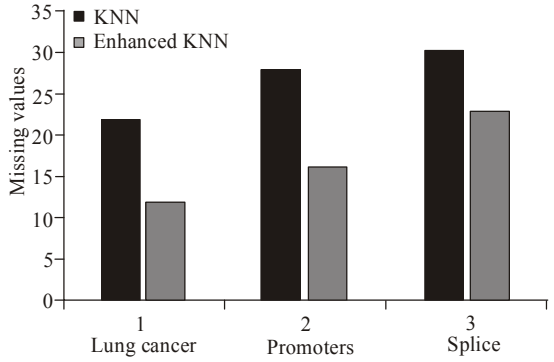

Fig. 5: Missing value detection using preprocessing process

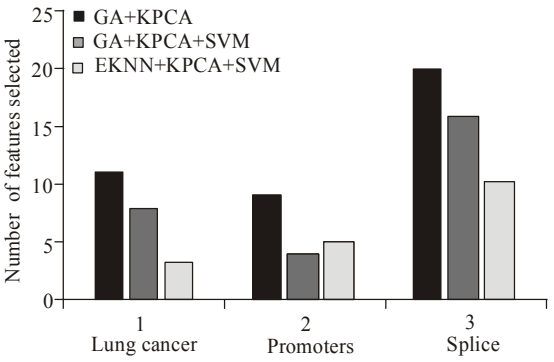

Fig. 6: Feature selection

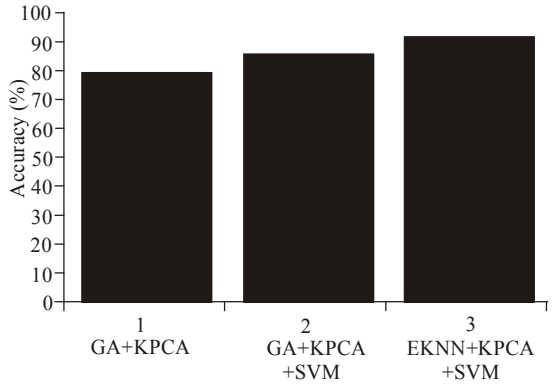

Fig. 7: Feature selection accuracy

Figure 7 shows the comparative analysis of feature selection accuracy of various approaches. It is to be noted that, the proposed Enhanced GA_KPCA SVM approach performs better and contain more accuracy than the other existing GA_KPCA and GA_KPCA SVM approaches.

Performance on classification process: The performance of the classification process is evaluated based on the parameters like:

- Average classification accuracy (%)
- Average convergence time period (CPU sec)
- Average Mean Square Error (MSE)

Table 5: Average classification accuracy in (%)

Techniques	Avg. classification accuracy in (%)
BPN	85.0
FKNN	93.2

Avg.: Average

Table 6: Average convergence time period

Techniques	Avg. convergence time period
BPN	16245
FKNN	15340

Avg.: Average

Table 7: Average Mean Square Error (MSE)

Techniques	Avg. mean square error
BPN	0.370
FKNN	0.257

Avg.: Average

Table 8: Average classification accuracy

Techniques	Accuracy in (%)
FKNN	93.2
ANFIS	97.4
FRB	94.2

Table 9: Average convergence time period

Techniques	Avg. convergence time period (CPU sec)
FKNN	15340
ANFIS	1540
FRB	10874

Avg.: Average

Table 10: Average Mean Square Error (MSE)

Techniques	Avg. Mean Square Error (MSE)
FKNN	0.410
ANFIS	0.151
FRB	0.250

Avg.: Average

For homogeneous classifier:

Average classification accuracy: Classification accuracy is the ratio of the total number of correctly classified large datasets to the total number of misclassified datasets. Average Classification Accuracy of existing BPN and the proposed FKNN is shown in Table 5.

Average convergence time period: Convergence time is a measure of how fast a group of routers reach the state of convergence. Average convergence time period of existing BPN and the proposed FKNN is shown in Table 6.

Average Mean Square Error (MSE): Average Mean Square Error (MSE) of existing BPN and the proposed FKNN is shown in Table 7.

On heterogeneous classifier:

Average classification accuracy: Table 8 shows the comparison results of Average classification accuracy for Heterogeneous ensemble classifiers.

Figure 8 shows the comparison results of Average classification accuracy for Heterogeneous ensemble classifiers. It is to be noted that, the proposed classification approach performs better and gives good results.

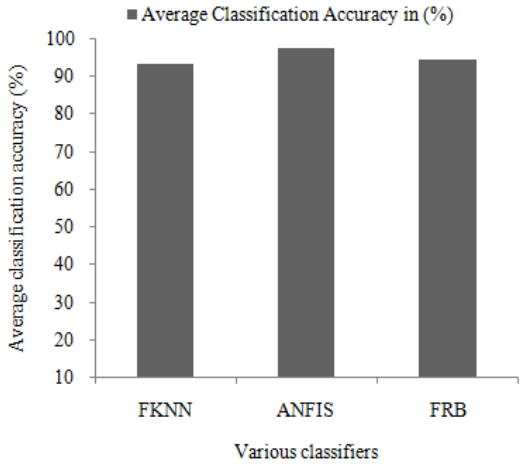

Fig. 8: Average classification accuracy graph for heterogeneous ensemble classifier

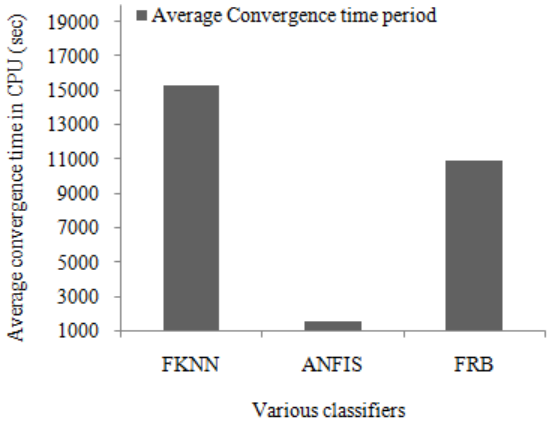

Fig. 9: Average convergence time graph for heterogeneous ensemble classifier

Average convergence time period: Table 9 shows the comparison results of Average convergence time for Heterogeneous ensemble classifiers.

Figure 9 shows the comparison results of Average convergence time for Heterogeneous ensemble classifiers. It is to be noted that, the proposed classification approach performs better and provides improved results.

Average Mean Square Error (MSE): Table 10 shows the comparison results of Average Mean square error for Heterogeneous ensemble classifiers.

Figure 10 shows the comparison results of Average Mean square error for Heterogeneous ensemble classifiers. It is to be noted that, the proposed

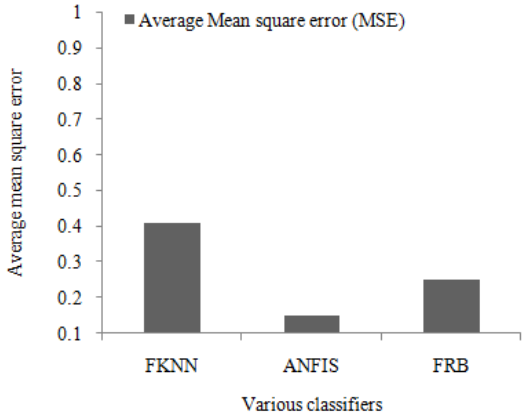

Fig. 10: Average mean square error graph for heterogeneous ensemble classifier

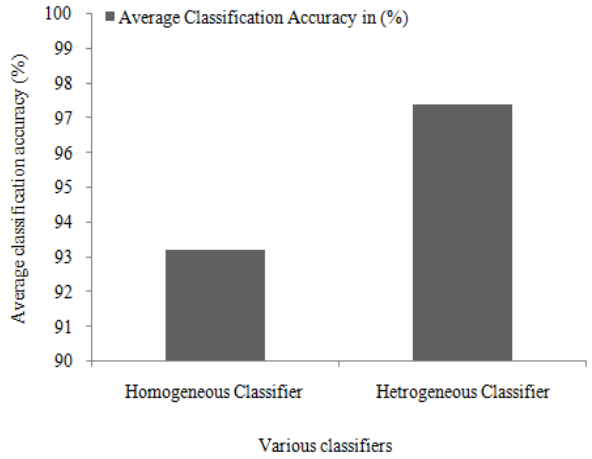

Fig. 11: Comparison of average classification accuracy

classification approach performs better and gives improved results.

Figure 11 shows the comparison results of Average Classification accuracy for Homogeneous and Heterogeneous ensemble classifiers. It is to be noted that, the proposed Heterogeneous ensemble classifiers gives better results than the homogeneous classifiers.

CONCLUSION

It is essential to remove the noisy and inappropriate features and data samples rooted in data sets before applying data mining techniques to examine the data sets. This study introduced a ensemble classification approach to classify the noisy and irrelevant features implanted in data sets and perceive the quality of the structure of data sets. In this study, an Enhanced KNN method is used as the preprocessing approach to find the missing values from the whole dataset. Then the feature selection of the datasets is processed using Enhanced Genetic Algorithm combined with Kernel PCA SVM Algorithm. Then, homogeneous and heterogeneous ensemble classification approaches are used in this research study for classification. In homogeneous ensemble classification model, Fuzzy KNN classifier is used. Then, heterogeneous ensemble classification framework is also proposed with the set of classifiers such as Fuzzy KNN, ANFIS and FRB. The performances of the both homogeneous and heterogeneous ensemble classification approaches are evaluated and it is observed that classification accuracy of heterogeneous ensemble classifier is comparatively higher than the homogeneous classification approach. Thus, the heterogeneous ensemble classifier performs better than the homogeneous ensemble classifier.

REFERENCES

Bin, N., J. Du, H. Liu, G. Xu, Z. Wang, Y. He and B. Li, 2009. Crowds' classification using hierarchical cluster, rough sets, principal component analysis and its combination. Proceeding of International Forum on Computer Science-Technology and Applications (IFCSTA'09), pp: 287-290.

Chen, H.L., D.Y. Liu, Y. Bo, L. Jie, W. Gang and S.J. Wang, 2011b. An Adaptive Fuzzy k-Nearest Neighbor Method Based on Parallel Particle Swarm Optimization for Bankruptcy Prediction. In: Huang, J.Z., L. Cao and J. Srivastava (Eds.): PAKDD. Part I, LNAI 6634, Springer Verlag, Berlin, Heidelberg, pp: 249-264.

Chen, H.L., Y. Bo, W. Gang, L. Jie, X. Xin, S.J. Wang and D.Y. Liu, 2011a. A novel bankruptcy prediction model based on an adaptive fuzzy k-nearest neighbor method. Knowl-Based Syst., 24(8): 1348-1359.

Cover, T.M. and P.E. Hart, 1967. Nearest neighbor pattern classification. IEEE T. Inform. Theory, 13(1): 21-27.

Dasarathy, B.V., 1990. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamos, CA.

Dhaliwal, D.S., P.S. Sandhu and S.N. Panda, 2011. Enhanced K-nearest neighbor algorithm. J. World Acad. Sci. Eng. Technol., 73: 681-685.

Ding, M., Z. Tian and H. Xu, 2009. Adaptive kernel principal analysis for online feature extraction. Proc. World Acad. Sci., Eng. Technol., 59: 288-293.

Fangjun, K., W. Xu, S. Zhang, Y. Wang and K. Liu, 2012. A novel approach of KPCA and SVM for intrusion detection. J. Comput. Inform. Syst., 8: 3237-3244.

Guha, S., R. Rastogi and K. Shim, 1998. CURE: An efficient clustering algorithm for large databases. Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data. Seattle, Washington, pp: 73-84.

Hart, P., 1967. Nearest neighbor pattern classification. IEEE T. Inform. Theory, 13(1): 21-27.

Hojjatoleslami, S.A. and J. Kittler, 1996. Detection of clusters of microcalcification using a k-nearest neighbour classifier. Proceeding of IEE Colloquium on Digital Mammography, pp: 10/1-10/6.

Jang, S.R., 1992. Self-learning fuzzy controllers based on temporal back propagation. IEEE T. Neural Networ., 3(5): 714-723.

Jang, S.R., 1993. ANFIS: Adaptive-network-based fuzzy inference system. IEEE T. Syst. Man Cyb., 23(3): 665-685.

Keller, J., 1985. A fuzzy k-nearest neighbor algorithm. IEEE T. Syst. Man Cyb., 15(4): 580-585.

Khan, J., J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson and S. Meltzer, 2001. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med., 7(6): 673-679.

Liao, W.Z. and J.S. Jiang, 2008. Image feature extraction based on kernel ICA. Image Signal Process., 2: 763-767.

Lior, R., 2010. Ensemble-based classifiers. Artif. Intell. Rev., 33: 1-39.

Pai, P.F. and W.C. Hong, 2005. Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms. Electr. Pow. Syst. Res., 74: 417-425.

Purnami, S.W., J.M. Zain and T. Heriawan, 2011. An alternative algorithm for classification large categorical dataset: K-mode clustering reduced support vector machine. Int. J. Database Theor. Appl., Vol. 4(1): 19-29.

Sanz, J., A. Fernándezb, H. Bustincea and F. Herrera, 2011. A genetic tuning to improve the performance of fuzzy rule-based classification systems with interval-valued fuzzy sets: Degree of ignorance and lateral position. Int. J. Approx. Reason., 52(6): 751-766.

Shipp, C.A. and L.I. Kuncheva, 2002. Relationships between combination methods and measures of diversity in combining classifiers. Inform. Fusion, 3: 135-148.

Somorjai, R.L., B. Dolenko and R. Baumgartner, 2003. Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: Curses, caveats and cautions. Bioinformatics, 19(12): 1484-1491.

Stefano, C.D., F. Fontanella and C. Marrocco, 2008. A GA-Based Feature Selection Algorithm for Remote Sensing Images. In: Giacobini, M. *et al*. (Ed.): Evo Workshops. LNCS 4974, Springer Verlag, Berlin, Heidelberg, pp: 285-294.

Yang, P., B. Zhou, Z. Zhang and A. Zomaya, 2010. A multi-filter enhanced genetic ensemble system for gene selection and sample classification of microarray data. BMC Bioinformatics, 11(Suppl 1): S5.