ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through
www.ijptonline.com

A NOVEL PRE-LEARNING ALGORITHM BASED DETECTION AND ELIMINATIONOF DATA ERROR AND DATA REDUNDANCY TO IMPROVE THE DATA QUALITY

[1]R.Mythily*, [2]W.Aisha Banus
[1]Department of Information Technology, B.S.Abdur Rahman Crescent University, Chennai.
[2]Department of Computer science and Engineering, B.S.Abdur Rahman Crescent University, Chennai.
Email: mythily@bsauniv.ac.in

Received on: 22-05-2017 *Accepted on: 28-06-2017*

Abstract

The volume of Big Data is rapidly increasing day by day, especially in the healthcare industry. This real world dataset has lots of error, redundancy and conflicts on the data which decreases the quality of data. The accuracy of data mining in any kind of dataset depends on the quality of data and it also affects greatly the performance of some classification algorithms in data mining. Not only, the cost is increased also the performance of the classifier is degraded by these noise parameters where they have to be removed in order to increase the efficacy and accuracy of the classifier. This process is named as data preprocessing or data cleaning or data tuning. In this paper, a Novel Pre-Learning (NPL) algorithm is proposed to improve the efficiency of data mining process. NPL comprises of four stages of tasks such as: (i). Learn the input data and eliminate the data-errors and data-redundancies. (ii). Reduces the data dimensionality by comparing the learned data with the original data. (iii). Utilizes Adaptive Feature Selection method for selecting appropriate features. (iv). Classify the data in terms of selected features using multi-class SVM. (v). finally the performance of the data classification is evaluated by comparing its results with the results of the other existing classifiers as SVM, Kstar, LWL, ID3 and J48. The result shows a significant improvement in the classification accuracy after error and redundancy removal. This experiment is carried out in MATALB software.

Keywords: Data Mining, Data Cleaning, Error Detection, Redundancy Removal, SVM classifier, Data Clustering and Classification.

Background Study: In recent days data generated and produced from various data-sources like websites, social networks, emails, stock exchange, share market and game, industries' reports, news groups. The type of the data is also different like text, images, audio, videos, and text-image, image-audio, text-audio and so on. According to the data type and amount of data, a number of intelligent information systems are created in order to arrange, cluster, classify, review, filter and generate reports on the data can be obtained by applying suitable information systems. The basic building block of information retrieval system is creating an intelligent system for categorizing documents physically and logically. Because of the data size increases in terms of electronic data, machine learning, supervised and unsupervised approaches are used for document categorization. Machine learning algorithms build a model from the training corpus by observing the characteristics of the documents under each category. Document representation is the primary step for any machine learning algorithm. Bag of Words model is the most widely adopted representation in document categorization [12]. Because of the high dimensionality most of the machine learning approaches is not able to perform well. Also, they are misrepresented due to redundant and irrelevant features in the database and sometimes it gives error message while triggering a query. To overcome this kind of issues the important features are gathered by applying a suitable feature selection method, where it helps to improve the classifier (Elena et al. 2005). A set of selected features are very common to all the data stored in the database to do efficient text categorization. A number of feature selection algorithms have been proposed and their efficiencies have been studied for text categorization. Some of the research works were utilized positive feedback based, distributed computation and constructive Meta heuristic approaches for efficient clustering and classification on any kind of data (Marco Dorgio et al. 2004).

Due to rapid development of the data mining, internet data, big data in terms of volume and in dimensionality there are various merging machine learning applications like text mining, image mining and information retrieval are emerging [1-3]. There are 16 trillion (means10^{13}) unique features are extracted using email-spam filter method which is discussed in [4]. Very high dimension of the data set needs more memory space, high computational cost whereas it can be reduced by dimensionality reduction. Various datasets produce with higher dimensions and most of the features are inessential to the output. Accordingly, falling the inessential features and chosen the most relevant features can immensely refute the generalization achievement. But in biomedical applications, it requires only less number of features to interpret the results for next biological analysis. From this it is very clear that any high dimensional problems and data sets need a sparse classifier for faster predictions and classifications. Non-Linear Machine learning approaches are mostly used for ultra-high dimensional data process. Some of the existing research works utilize linear techniques for feature mappings in order to tackle the intrinsic non-linearity data [5-6]. Feature mapping algorithms always increase the dimensionality of the data. So, dimensionality reduction becomes more important and few feature mappings follows spectrum-based feature mapping for string data, and histogram based kernel feature for mapping image data [7].

Various feature selection method has been proposed for classifying big data in the past decades [8-9]. Generally the feature selection methods are divided into two categories such as Filter methods and Wrapper methods [9]. Signal to noise ratio method, spectral feature filtering method are best in lowest computational cost, but they are not capable for finding best feature subset according to a predictive model. Opposite to that, wrapper methods including learning rules they can select relevant features [10 -11], but wrapper methods are computationally expensive than filter methods. Since here it is focused to utilize wrapper methods for analyzing big data. In recent real time applications like photo sharing, social networking and video sharing are utilizing enormous volumes of all kinds of data, which is available to aid in decision making. One of the effective methods is feature selection in classification of big data in fields such as data mining, pattern matching, pattern recognition [1], bio-information [2], arrhythmia classification [3] and numerous others. Supervised learning methods like decision trees [4], support vector machines (SVM) [5-7] and neural networks [8-11] are used to classify data into appropriate categories.

From the above background study it is understand that yet there is no methodology can fulfill the efficiency in terms of clustering and classification. Since this paper motivated to provides an efficient methodology for improving the classification and providing relevant result for queries.

Preliminaries

Before internet collecting data from various sources in time is very difficult and too expensive. In general the data is under the control of the distribution center and any permissible data owner can get the data after verification. Information mining, the extraction of concealed prescient data from substantial databases, is an intense new innovation with incredible potential to help organizations concentrate on the most critical data in their information stockrooms. Information mining instruments foresee future patterns and practices, permitting organizations to make proactive, learning driven choices. Information mining apparatuses can answer business addresses that generally were excessively tedious, making it impossible to determine. They search databases for hidden designs, finding perceptive data that specialists may miss since it lies outside. After a long process and product development data mining techniques become useful and necessary to the computing industries. Information mining strategies are the consequence of a long procedure of exploration and item improvement. This advancement started when business information was initially put away on PCs, preceded with enhancements in information access, and all the more as of late, produced advances that permit clients to explore through their information progressively. Information digging is prepared for application in the business group since it is upheld by advances that are presently adequately develop: Definition of Data Mining Data mining is the procedure of investigation and examination via programmed or self-loader method for vast and examination ties of information keeping in mind the end goal to find importance full examples and guidelines – M.J. Berry and G.S. Linoff. Information Mining Process a definitive objective of information mining is expectation and prescient information mining is the most widely recognized sort of information mining. The procedure of information mining comprised of three phases:

- Preprocessing the Data
- Dimensionality Reduction
- Feature Selection
- Classification
- Performance Analysis

Problem Statement

Data mining and knowledge discovery is one of the drastically growing areas in computing environment. Incoming data, data size, type of data and streaming speed are changing in increased manner due to industry growth. Identifying and fetching a peculiar data with more accuracy is too difficult due to the nature of the data and characteristics of the data. Because of different data sources, types and data preparation ways has more error in the data and duplication in the data. During the query process, data error and redundancy doesn't provide relevant result. For example, a data entity value is **"$%^#$RFEE@#097y**" (unknown data value), during query process the DBMS cannot compare the values of the particular field and it provides error message. In a big organizations like database based application running industries error in the data and duplicate data are gives error message during query process. In order to correct the data without error and redundancy initially learn the data and remove the error and eliminate the redundancy. This problem is taken into account and a novel pre-learning approach combined with multi-class SVM to increase the efficiency of the data mining and it is described in detail below. The problem used binary classification for feature selection.

Let $\{(\boldsymbol{x_s}, \boldsymbol{y_s}) | \boldsymbol{s = 1}, \dots, \boldsymbol{S}\}$ is a sequence of input patterns received over the trials, where each $\boldsymbol{x_s} \in \mathbb{R}^{\boldsymbol{d}}$ is a vector having a dimension of d and $\boldsymbol{y_s} \in \{\boldsymbol{-1}, \boldsymbol{+1}\}$. The dimension d is assumed as a large number, due to reduce the computational efficiency it is essential to select a relatively small number of features for linear classification. In each sequence s, the learner provides a classifier $\boldsymbol{w_s} \in \mathbb{R}^{d}$ is used to classify the data $x_s$by a linear function sign **($\boldsymbol{w_s^S x_s}$).** Using all the features take more time, we take w_s to have at most B non-zero elements, i.e.

$$||\boldsymbol{w_s}||\boldsymbol{0} \leq \boldsymbol{B}$$

Where, **B > 0** is a predefined constant, and **B** features of $\boldsymbol{x_s}$ is used for classification. In this paper the main objective is to design an efficient strategy for feature selection which is able to reduce all kind of mistakes while query process and in the entire paper it is assumed that$||\boldsymbol{X_s}||\boldsymbol{2} \ \leq \boldsymbol{1}, \boldsymbol{s} = \boldsymbol{1}, \dots, \boldsymbol{S}$**.start**

Preprocessing by Novel Pre-Learning algorithm

The data sources are different and it may consist of errors and redundancy. Any data manipulation methods do not provide results due to error and redundancy occurrence in the data. Here the initial process of NPL is learning the data for feature selection. Before going to start the learning process on the data, it is essential to clear the data by removing the noise and data redundancy. The set of all data **X** is verified by checking the data size, data format (i.e. data type) and it can be done by:

$$\begin{Bmatrix} \boldsymbol{X(i) = -1} \qquad \boldsymbol{if\ (datatype\big(X(i)\big) is\ different\ than\ (X)} \\ \end{Bmatrix}$$

Where, -1 denoted that the data is having error. Similarly the data redundancy can be verified and eliminated by:

$$\{\boldsymbol{X(i) = \ -1 \quad if\ (X(i) == X}\ \}$$

Whenever one piece of data is matched with the other data in the entire dataset it will be marked as -1 (means it is eliminated). A Novel Pre-Learning (NPL) algorithm is used to learn the dataset used for experiment but due to reduce the computational complexity only a small % of the original data is taken and learn the data.

$$\textbf{Let X = \{ } \mathbf{X_1, X_2, \dots, X_n} \textbf{\}}$$

Where n is the size of the dataset. A portion of the dataset is used for pre-learning is

$$\textbf{TP =\{}\mathbf{X_1, X_2, \dots X_{20}}\textbf{\},}$$

$$\boldsymbol{TP\ \subseteq X}$$

Means 20% of the data is taken for learning due to reduce the computational complexity, computational time and memory usage. The learned attributes of the TP is

$$\mathbf{F(TP(X_i)) = \{f_1, f_2, \ldots, f_k\}}$$

$$\mathbf{F(TP)} \subseteq \boldsymbol{F}$$

After using a feature selection method

$$\mathbf{fs(F) = f_1 \cap f_2 \cap f_3, \ldots\ldots, \cap f_k}$$

an optimal subset of features are chosen according the objective functions where the user going to apply. Main objectives are reducing the dimensionality, removing noise and improving the mining performance by learning the speed, predictive accuracy and simplicity of mined results. NPL retrieves a set of features from **TP(X_i)** and obtain a common feature set for the entire data **TP(X).** The entire process of feature selection for the entire process is depicted in the following Fig-1.

Dimensionality Reduction

Dimensionality reduction makes easy and convenient to collect the data for experiment. Data collection can be done to accumulate in an unprecedented speed. Also to improve the speed dimensionality reduction will downsize the data.

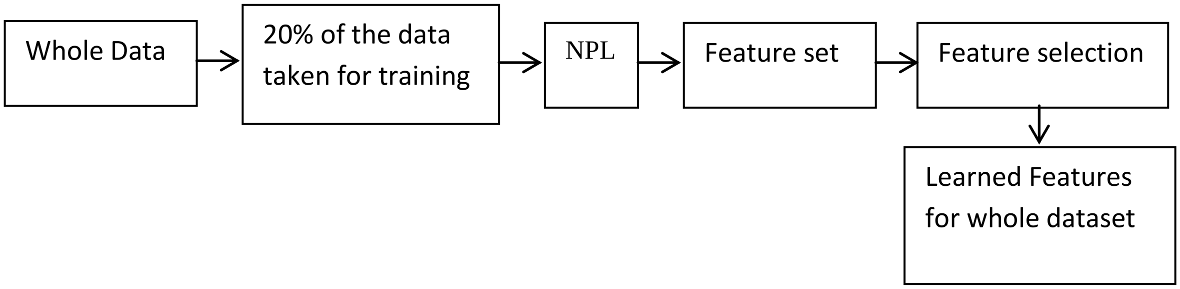

Whole Data
20% of the data taken for training
NPL
Feature set
Feature selection
Learned Features for whole dataset

Fig-1: Feature Selection.

The volume of information that is being made as public, is expanding each year quickly and information produced by machine is more contrasted with the information created by human. Naturally information is created from machine logs, RFID readers, sensor systems, GPS vehicle follows, and so on. To contend in the information economy, commercial enterprises need to oversee information from outer sources as well, aside from dealing with their own. It has been watched that it is conceivable to beat better calculations by having more information. As of late the capacity limits of hard drives have expanded massively, however the entrance rate has not been kept up. Composing information to hard plate is still slower when contrasted with perusing. The conceivable answer for this issue is to circulate the information among numerous machines and working in parallel.

The principal challenge in data mining is to reduce the data dimension in order to avoid computational complexity. This issue is tended to by the Map Reduce structure by having various duplicates of information. The following test associated with disseminated preparing is to join information from various machines. The Map Reduce structure gives an exceptional record framework and a programming model that covers the issue of perusing and composing information by changing those operations to a calculation of an arrangement of keys and values. MapReduce projects are naturally parallel that makes huge scale information examination basic with various machines of sensible designs. In this programming model information is handled as a rundown of keys and values. Guide and diminish are the two vital capacities in this programming model. Map capacity takes a key and a worth and returns a rundown of keys and values. The information sort might be not the same as the yield information sort. The dimensionality reduction in this paper is depicted in Fig-2. The entire features are extracted, filtered and categorized the features according to the uniqueness. Finally the squared feature data is converted into a vector. The Map Reducing functionality reduces the dimensionality by mapping the learned features with the available features in the original dataset. The way the dimensionality reduction is shown is Fig-2.

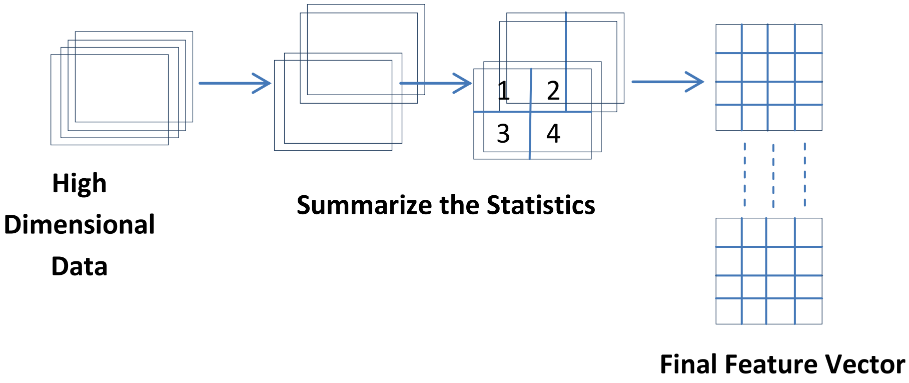


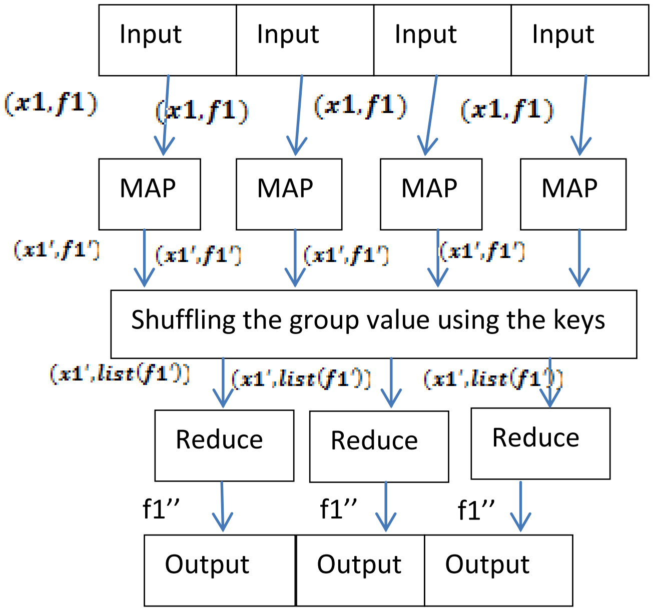
Fig-2: **Dimensionality Reduction.**

NPL reduces the high dimension into countable one, comparing the learned information with entire dataset. The contribution for the decrease capacity is a key and a rundown of qualities relating to that key. The yield of lessen capacity is a rundown of prepared qualities relating to the info key. The information and yield of the mapper and reducer capacity is given as beneath:

$$\boldsymbol{map(X1, f1) \rightarrow list(X2, f2)}$$

$$\boldsymbol{reduce\big(X2, list(f2)\big) \rightarrow list(f2)}$$

It might be seen from the scientific representation of guide and lessen capacities, that the yield sorts of mapper and information sorts of reducer must be indistinguishable. The information to be handled is put away in the appropriated record framework. The document is split into various parts by the MapReduce structure and given to mapper assignments. Mappers run the code on the information part and store the moderate results on the nearby machine. The guide yields are moved to the spot of reducer to prepare. This procedure is termed as rearranging. Since the transmission capacity is restricted the MapReduce system streamlines by giving one more capacity called combiner.



$$\boldsymbol{map}(\boldsymbol{X1}, \boldsymbol{f1}) \rightarrow \boldsymbol{list}(\boldsymbol{X1'}, \boldsymbol{f1'})$$

$$\boldsymbol{reduce}\left(\boldsymbol{X1'}, \boldsymbol{list}(\boldsymbol{f1'})\right) \rightarrow \boldsymbol{f1''}$$

Fig-3: Map Reducing.

By and large the combiner capacity works on the yield of the Mapper assignments and produce yields of lessened size. In the greater part of the cases the combiner capacity is same as the lessen capacity. Complex issues are fathomed utilizing MapReduce by having more MapReduce employments, as opposed to having complex MapReduce capacities. MapReduce reduces the size of the data in terms of features. The dimensionality reduction by map reducing is proved already in various research works. The step by step map reducing process is depicted Fig-3. Dimensionality reduction is also shown in Theorem-1.

Theorem-1: Using Saddle-point theorem the following equality holds by interchanging the order of min(d∈D) and max(α∈Att) in minmax d∈D, α∈Att (-f(α, d).

$$\min\max_{d \in D,\, \alpha \in Att} (-f(\alpha, d)) = \max\min_{d \in D,\, \alpha \in Att} (-f(\alpha, d))$$

Algorithm-1: Novel_Pre-Learning Algorithm

Given ***X*** *is the whole data,* $\boldsymbol{X_i}$*is the subset,* ***n*** *is number of sub-dataset,* ***F*** *is the feature array and feature mapping* ***map(x)***

```
X is divided into {$X_1$, $X_2$,…, $X_n$}
for i=1 to n
        learn the feature $f_i$ with respect to $X_i$ according to map(x)
            arrange $f_i$ and update it in F
            for j =i+1, …, n
            calculate the features with respect to $f_i$ for $X_1$ to $X_n$
            FF = $f_i \cap f_j$
    end
end
return FF
```

Feature Selection

In this paper in order to improve the efficiency of the proposed approach it is essential to concentrate on the feature selection process taken from [30]. It is well known that NPL learned a portion of the data and provides a set of labeled patterns as$\{\boldsymbol{x_i}, \boldsymbol{y_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{n}}$, where$\boldsymbol{x_i} \in \mathbb{R}^{\boldsymbol{m}}$ is the data having $\boldsymbol{m}$ number of features and $\boldsymbol{y_i} \in \{\pm\boldsymbol{1}\}$ is the output label. In order to avoid the over-fitting problem people usually introduce certain regulations to the loss function. For $\boldsymbol{x_i}$ to select features which contribute the most to the margin, it can learn a sparse decision function $d(\boldsymbol{x}) = \boldsymbol{w'x}$ by solving:

$$\min_{\boldsymbol{w}} \|\boldsymbol{w}\|_{\boldsymbol{0}} + \boldsymbol{C} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{l}(-\boldsymbol{y_i w' x_i})$$

Where$l(\cdot)$ is the loss function, $w \in \mathbb{R}^m$ is the weight vector, $\|w\|_0$ is the $l - norm$ which counts the number of non-zeros in$w$, and $C > 0$ is the regulation parameter. In this paper NPL apply a feature scaling for feature selection by introducing a continuous feature scaling vector$d \in [0,1]^m$. In order to force the sparsity, it is imposed an explicit $l - norm$constraint$\|d\|_1 \leq B$, where the scalar B denotes the least number of features to be selected. For all the data entries X, the set of features F is selected where $f^i \in \mathbb{R}^m$ denotes the feature vector selected and verified as zero vector or one vector as $0 \in \mathbb{R}^m$ and $1 \in \mathbb{R}^m$ respectively.

Multi-Class SVM approach

Support vector machines (SVM) were initially intended for twofold arrangement. Step by step instructions to successfully broaden it for multi-class arrangement are still an on-going examination issue. A few strategies have been proposed where ordinarily we develop a multi-class classifier by consolidating a few twofold classifiers. Some creators likewise proposed strategies that consider all classes without a moment's delay. As it is computationally more costly to illuminate multiclass issues, correlations of these techniques utilizing substantial scale issues have not been genuinely led. Particularly for strategies settling multi-class SVM in one stage, a much bigger advancement issue is required so up to now tests are restricted to little information sets [15].

SVM and Multi-Class SVM are belongs to machine learning approaches, it is able to learn any kind of data by itself without any external technical supports. Both approaches has in-built mechanisms to verify the similarities, distance and matching score among two different objects such as trained object and test object.

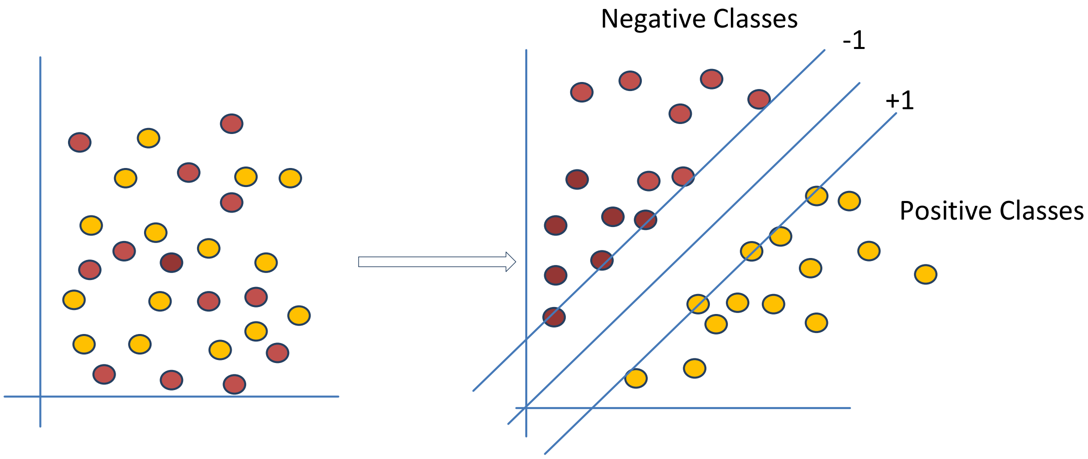


Fig-4: **Support Vector Machine/Optimal hyper plane.**

The primary function of SVM finds the optimal hyper plane and it is used for classifying the data which is shown in Fig-4. The Figure shows that the optimal has two different margins representing two different classes such as {-1} and {1}. The circles sitting on the hyper plane shows that it is the highest optimum value achieved by the SVM and it is the highest classification accuracy. SVM classifier classifies the data linearly as:

$$\boldsymbol{f^k : X^d \rightarrow \{-1, +1\}}$$

In the form of

$$\boldsymbol{\hat{f}^k(x) = sgn\left(g^k(x)\right) = sgn(x^T w^k + b^k)}$$

Where,

w is the weight of **x**, and $\|\boldsymbol{w^k}\|_2$ **is 1,** $\boldsymbol{g^k(x)}$ gives the Euclidean distance from**x** to the feature boundary$\boldsymbol{f^k}$. SVM minimizes the mean square error over the sample data among the normalized data and the target data lies between$y \in \{\boldsymbol{-1, +1}\}$**.** -1 and +1 are the two different classes separated from the original dataset.

Algorithm for SVM Classifier:

```
/**
*This algorithm solves the classifier problem by calling
* Set of points in the SV array.
*
*Input: Input data
*Output: Set of support Vectors Extracted from sample data
*Initialization: Error threshold = huge value **/
begin
        Randomly sample 2 pixels belong to different classes.
        Add them to current set of SV
        Set the corresponding variables ('α') values
        Loop SVM
          Loop to randomly consider other samples
            Choose the set of points with which current SV gives sample point less than the current error threshold.
         End loop random sample some other pixel
        Update error threshold as average of sampled test errors.
        Loop over misclassified points
          Add the point to current SVs
          Train the data set over the remaining pixels.
        End loop over misclassified points
        Save variable ('α') for next iteration
        End Loop SVM
End
```

But the Multi-class SVM classifies the data under various classes required by the user.

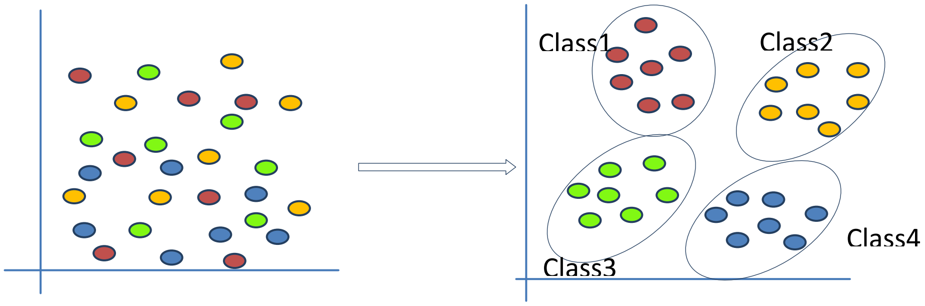


Fig-5: Multi Class SVM.

The above Fig-4 and Fig-5 illustrated the classification by SVM and multi-class SVM. SVM classifies the entire data in to two different classes as -1 and +1. But the multi-class SVM classifier classifies the entire data into various classes according to the user defined classes like {-1, 0, +1...}.Multi SVM classifier classifies the data linearly as:

$$\boldsymbol{f^k}: \boldsymbol{X^d} \rightarrow \{\boldsymbol{-1}, \boldsymbol{0}, \boldsymbol{+1}, \ldots.\}$$

Due to various classes available in the input data, here multi-class SVM is applied for classification.

Experimental Results

In this paper an extensive experiment is conducted to evaluate the performance of the proposed novel pre-learning process based feature selection algorithm. Here it is evaluated the NPL performance of the learning and feature selection process on various benchmark datasets taken from UCI repository and demonstrate the proposed approach discussed in this paper. Also the obtained results are compared with the other learning techniques and feature selection approaches combined with examining scalability verification in large size dataset. In order to examine the performance of the proposed approach there are numerous public datasets is taken from web machine learning repositories. All the dataset is downloaded from LIBSVM [16] and UCI machine learning repository [17]. The following Table-1 shows a number of datasets taken from UCI repository used for experiment the proposed approach. There are five data sets with different number of records, different features, different classes and from different fields are given in Table.-1.

Table-1: UCI Dataset Used for Experiment.

Dataset	Number of Classes	Number of Instances	Number of Features
Vowel	11	530	10
Vehicle	4	850	18
Robot	4	5456	24
Glass	6	215	9
Bupa	2	350	6
Total	27	7401	67

In order to compare the effectiveness of the proposed approach there are three different experiments is carried out. One is the dataset without features, other one is dataset after feature selection using NPL final one is dataset after feature selection using SVM. The main objective of the paper is to do feature selection for improving the mining accuracy in a data set. In order to do this a sequence of data mining procedures are applied in this paper. Initially, here the experiment begins from data cleaning. There are two different process is done by data cleaning and it is done by NPL method. In order to calculate the efficiency initially the data cleaning is calculated against the total number of instances.

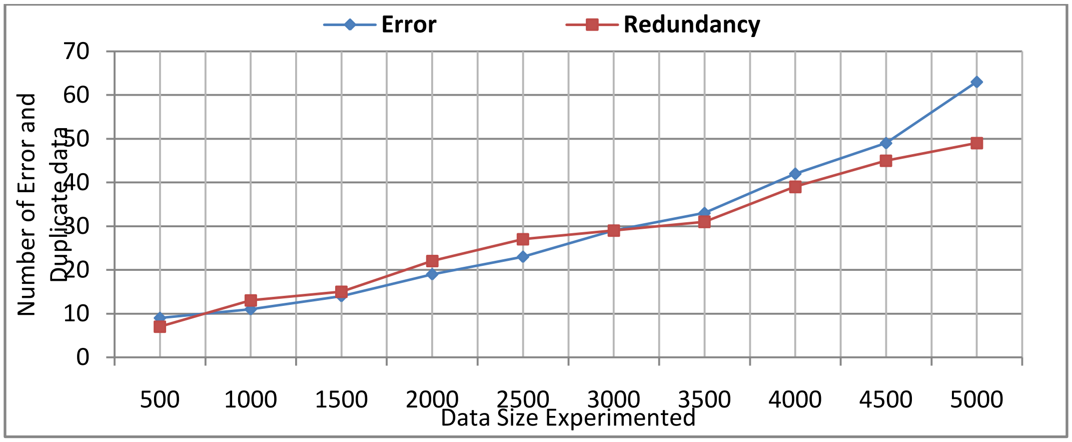


Fig-6: Availability of Error Data and Redundancy Data.

Fig-6 shows the experimental results of the data cleaning process. The number of error in the data is depending on the number of total data deployed. If the size of the increases the error occurred in the data is also getting increased, since if it is less number of data people are careful with converting into digital data. But if the data size is growing in large volume then digital data creates some errors (while conversion) automatically. Error can be created automatically or manually without human knowledge and it is because of data format, comes from various sources and speed of the data streamed in the network. In this paper the number of error and redundancy against the total number of taken for experiment is calculated and the result is shown in Fig-6. The second level process of NPL is dimensionality reduction. Reducing the dimension leads to reduce the number of comparisons and computational time. If the data is high dimensional data then the time taken for learning the data, cleaning the data and time taken to learn, feature selection are become too complex it takes too much amount of time. Finding a similarity between two different data can be obtained by comparing two element of the dataset that is two data in the entire set. Comparing pair-pair of data in the entire data set takes too much of time, whereas the efficiency can also be calculated using the computational time. In this paper, the time taken to calculate the dataset is calculated for finding the time complexity.

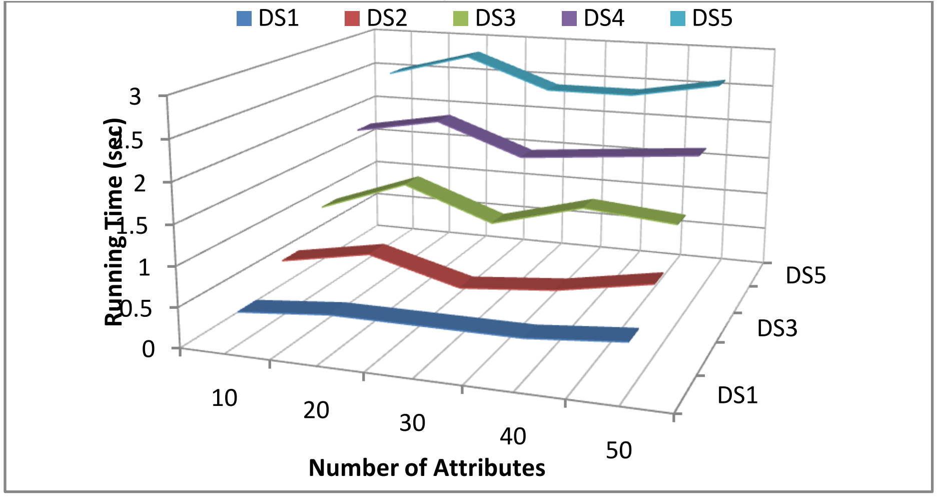


Fig-7: Attributes versus Time.

Comparison is carried out over the set of attributes on the dataset. Initially pair of data is collected in same attribute based and compared to find the similarity. Fig-7 shows the time duration taken for selecting pair of data from same attribute groups and compared. It is clear from Fig-7 the time taken for pair wise comparison is different for different attributes. The time taken for comparison is increased when the number of attributes increased. Also the time taken depends on the dataset also since the attribute and data behavior is different for various data types. Before going to compare the data pairs, it is essential to separate data in each data set in terms of attributes. To do that, the entire datasets are applied for clustering process whereas the clustering process groups the data against the attributes (properties). In this paper in order to check the performance of the proposed approach the accuracy of the clustering process is calculated by comparing the number of clusters available and clusters obtained automatically by the proposed approach. Here the dataset taken for experiment is clustered benchmark data verified in the existing research works [31].

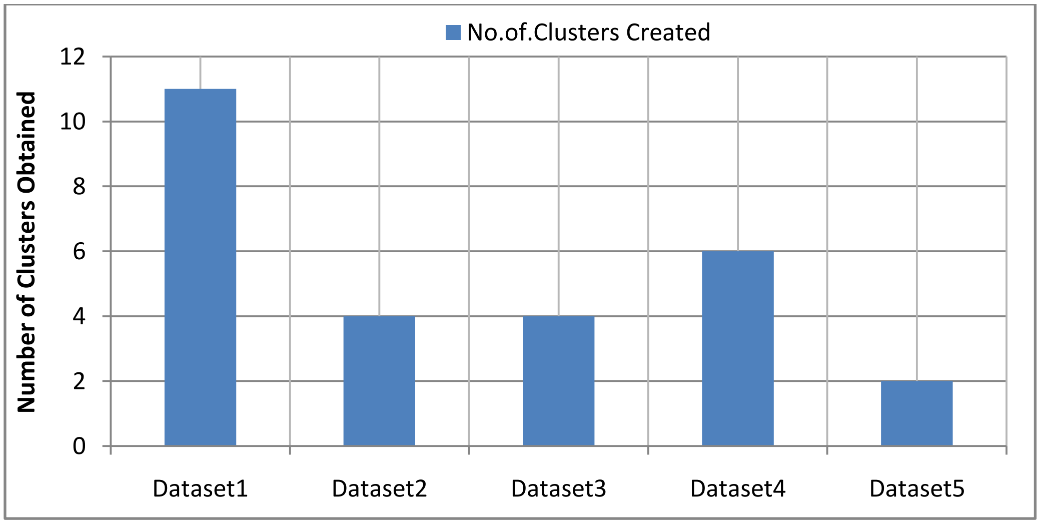


Fig-8: Number Clusters Obtained in the Experiment.

The number of clusters obtained automatically by the proposed approach is merely equal to the available clusters in the dataset. Also the number of clusters available and obtained is different for different datasets. Dataset-1 contains more clusters than the other datasets 2 to 5. Fig-8 shows the number of clusters obtained automatically from various datasets given in Table-1. Once the clustering efficiency is verified, then the features are extracted and best-common features are selected for mining process. The efficacy of mining a data from a large set of data is depending on clustering and classification on features. In this paper the features are extracted and selected for improving the accuracy is experimented and the obtained result is given in Fig-9. It is well known that a large size data consists of more number of features (attributes / properties) by nature. For example, a man might have a name, height, weight, color nationality and etc. But for classification common attributes are chosen for grouping more data into a single group. For example sports, talent and cultural behavior cannot bring more number of data together and it eliminates the data from the mining process. So, it is necessary to select features from the extracted features whereas it can group more number of data into a single cluster or group. In general the numbers of available features are more than the other extracted and selected features in a dataset. It is clearly depicted in Fig-9 that the number of existing features are high, the number of extracted features are little lesser than the available features, but the number of features selected are lesser than the others. For example in dataset-1, the number of available features is 110, the extracted features are 79 and the number of selected features is 33. This process of feature extraction and classification is applied on the training data and testing data separately. Testing process based features are given in Fig-9 and the accuracy of testing is calculated by the feature selection process on the testing data.

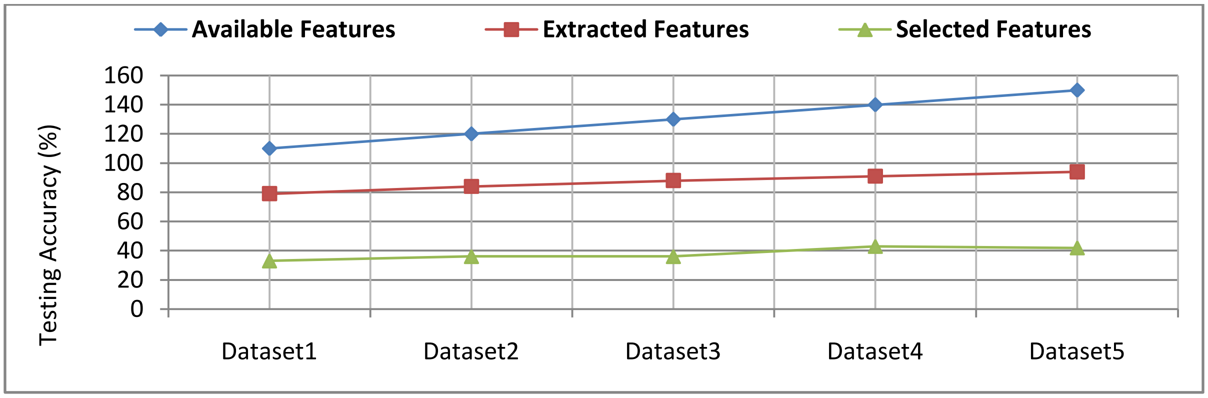


Fig-9: Testing Accuracy Calculation in Various Datasets.

After feature selection the proposed approach proceed with the dimensionality reduction by mapping the features. Map reduction is used to reduce the dimensionality of the dataset where it is used to reduce the time complexity and computational complexity. Pair of data or clusters for data is taken and their features are mapped using $\boldsymbol{MAP}(\boldsymbol{x},\boldsymbol{f})$ function and it calculates the mapped data pair and reduce the size of the data. For example the two data having same features are linked by a single pointer. For example in dataset-1, the selected number of features is 33 whereas the mapped features are 13.

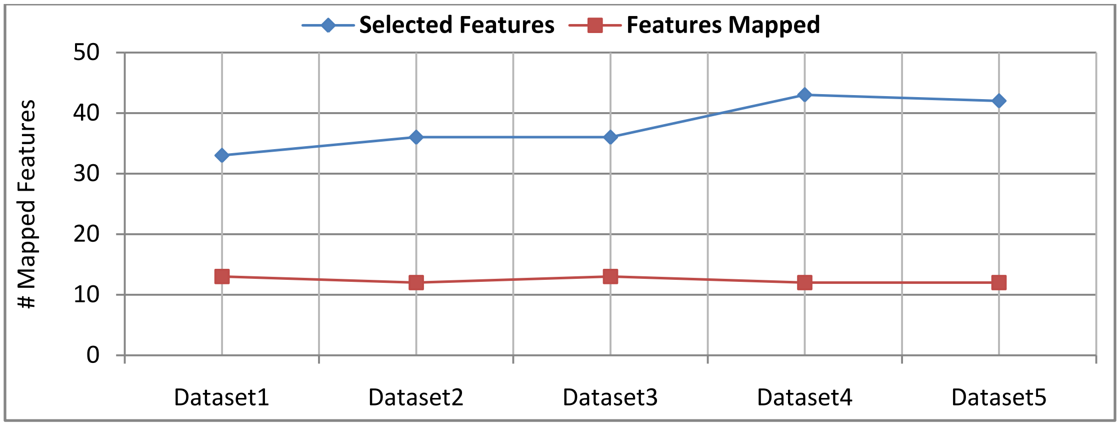


Fig-10: Feature Mapping On various Dataset.

So instead of persisting 33 data label in memory it can give only 13 data labels which reduce the memory utilization, comparison process and time complexity. This process is experimented using our proposed approach and the obtained result is given in Fig-10. Fig-10 shows clearly that the number of mapped features is very less than the selected features where it leads to improve the efficiency. In order to reduce the computational complexity by comparing the data map reducing is utilized. But in order to reduce the time complexity the training time is calculated. The time taken to train the data in terms of number of features is increased when the number of features increased. Fig-11 shows the time taken for training the data in terms of features. It is clear from the figure that when number of feature increased then the amount of time taken for training is increased.

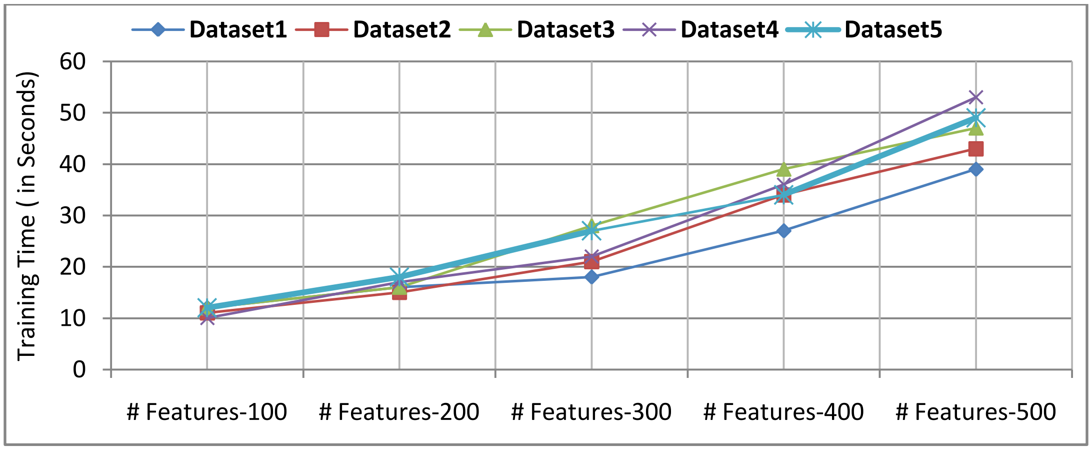


Fig-11: Performance Evaluation in Terms of Training Time Computation.

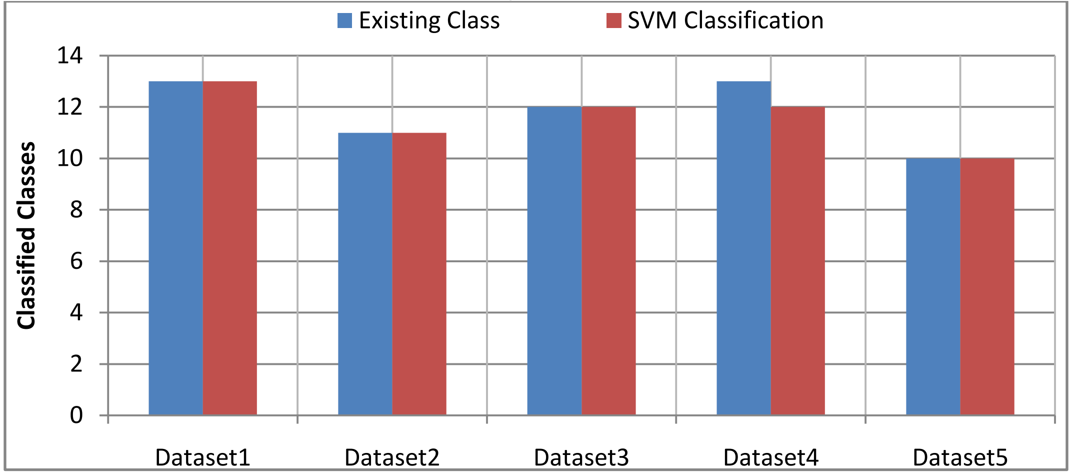


Fig-12: SVM Classifier vs. Available Classification.

Finally the classification accuracy is calculated to evaluate the performance of the approach. This can be calculated by comparing the existing classified results with the SVM classifier results. Classification is one of the methodologies of labeling the data in the dataset. The SVM classifier result is merely equal to the existing classified results in the database and it is shown in Fig-12. For example, Fig-12 says that for dataset-1, the number of classified data and the SVM classified data are 13. From the experimental results it is clear that the proposed approach is efficient than the existing approach and it suits for various datasets.

Conclusion

The main objective of this paper is to provide an efficient approach for data preprocessing and dimensionality reduction approach for large set of data. A Novel pre learning algorithms is applied here to learn the data for dimensionality reduction in terms of features. NPL clear the error, eliminate the data redundancy and extract feature, select features then apply dimensionality reduction by mapping the features. Since this NPL approach apply various data mining process sequentially the efficacy of the approach is improved the quality of the data in a better level than the existing approaches discussed in the literature. Also from experimental results it is proved that NPL algorithm is suitable for cleaning and classifying the various kinds of large datasets.

References

1. J. Deng, A. C. Berg, and F. Li, “Hierarchical semantic indexing for large scale image retrieval”, In CVPR, pages 785-792. IEEE, 2011.
2. P. Li, A. Shrivastava, J. Moore, and A. C. Konig, “Hashing algorithms for large-scale learning”, In NIPS, 2011.
3. P. Li, A. Owen, and C. H. Zhang, “One permutation hashing”, In NIPS, 2012.

4. K. Weinberger, A. Dasgupta, J. Langford, A. Smola, and J. Attenberg, “Feature hashing for large scale multitask learning”, In ICML, 2009.

5. Y. W. Chang, C. J. Hsieh, K. W. Chang, M. Ringgaard, and C. J. Lin, “Training and testing low-degree polynomial data mappings via linear SVM”, JMLR, 11:1471-1490, 2010.

6. S. Maji and A. C. Berg, “Max-margin additive classifiers for detection”, In ICCV, 2009.

7. J. Wu , “Efficient HIKSVM learning for image classification”, IEEE Trans. Image Process, 21(10):4442-4453, 2012.

8. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, “Gene selection for cancer classification using support vector machines”, Machine Learning, 46:389-422, 2002.

9. O. Chapelle and S. S. Keerthi, “Multi-class feature selection with support vector machines”, In Proceedings of the American Statistical Association, 2008.

10. Z. Xu, R. Jin, Ye J., Michael R. Lyu, and King I, “Non-monotonic feature selection”, InICML, 2009a.

11. I. Guyon and A. Elisseefi, “An introduction to variable and feature selection”, JMLR, 3: 1157-1182, 2003.

12. Huan Liu (2005), “Evolving Feature Selection”, *IEEE Intelligent Systems, Volume 20, Issue6,* 64-76.

13. A. Nedic and A. Ozdaglar, “Sub Gradient methods for saddle-point problems”, Journal of Optimization Theory Application., 142(1):205-228, 2009.

14. A. Nemirovski, “Prox-method with rate of convergence o(1/t) for vibrational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems”, SIAM J. Opt., 15:229-251, 2005.

15. Hsu, Chih-Wei, and Chih-Jen Lin. "A comparison of methods for multiclass support vector machines." *Neural Networks, IEEE Transactions on* 13.2 (2002): 415-425.

16. http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/

17. http://www.ics.uci.edu/~mlearn/MLRepository.html

18. http://www.cs.toronto.edu/~kriz/cifar.html

19. J. K. Kishore, L. M. Patnaik, V. Mani, and V. K. Agrawal, “Application of genetic programming for multicategory pattern classification,” *IEEE Transactions on Evolutionary Computation,*vol. 4, no. 3, pp. 242–258, 2000.

20. R. Stevens, C. Goble, P. Baker, and A. Brass, “A classification oftasks in bioinformatics,” *Bioinformatics*, vol. 17, no. 2, pp. 180–188, 2001.

21. E. Namsrai, T. Munkhdalai, M. Li, J.-H. Shin, O.-E.Namsrai,and K. H. Ryu, “A feature selection-based ensemble methodfor arrhythmia classification,” *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 31–40, 2013.
22. J. R.Quinlan, *C4.5: Programs for Machine Learning*, The Morgan Kaufmann Series in Machine Learning, Morgan Kaufmann, San Mateo, Calif, USA, 1993.
23. T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M.Schummer, and D. Haussler, “Support vector machine classificationand validation of cancer tissue samples using microarray expression data,” *Bioinformatics*, vol. 16, no. 10, pp. 906–914,2000.
24. Y. S. Hwang, J. B. Kwon, J. C.Moon, and S. Je Cho, “Classifying malicious web pages by using an adaptive support vector machine,” *Journal of Information Processing Systems*, vol. 9, no.3, pp. 395–404, 2013.
25. J. Uddin, R. Islam, and J. Kim, “Texture feature extraction techniques for fault diagnosis of induction motors,” *Journal ofConvergence*, vol. 5, no. 2, pp. 15–20, 2014.
26. G. P. Zhang, “Neural networks for classification: a survey, ”*IEEE Transactions on Systems, Man and Cybernetics Part C:Applications and Reviews*, vol. 30, no. 4, pp. 451–462, 2000.
27. M. Malkawi and O. Murad, “Artificial neuro fuzzy logic system for detecting human emotions,” *Human-Centric Computing and Information Sciences*, vol. 3, article 3, 2013.
28. A. K. Gopala krishna, “A subjective job scheduler based on aback propagation neural network,” *Human-Centric Computing and Information Sciences*, vol. 3, article 17, 2013.
29. H. Lee, H.Kim, and J. Seo, “An integrated neural network model for domain action determination in goal-oriented dialogues,”*Journal of Information Processing Systems*, vol. 9, no. 2, pp. 259–270, 2013.
30. Mingkui Tan, Ivor W. Tsang and Li Wang, “Towards Ultrahigh Dimensional Feature Selection for Big Data”, Journal of Machine Learning Research 15 (2014) 1371-1429.
31. Kuan-Cheng Lin, Sih-Yang Chen and Jason C. Hung, ”Feature Selection and Parameter Optimization of Support Vector Machines Based on Modified Artificial Fish Swarm Algorithms”, Hindawi Publishing Corporation, Mathematical Problems in Engineering, Volume 2015, Article ID 604108, http://dx.doi.org/10.1155/2015/604108.