Complex Intell. Syst. (2018) 4:105–118
https://doi.org/10.1007/s40747-017-0060-x

ORIGINAL ARTICLE

EFS-MI: an ensemble feature selection method for classification

An ensemble feature selection method

Nazrul Hoque[1] · Mihir Singh[2] · Dhruba K. Bhattacharyya[2]

Received: 3 February 2017 / Accepted: 24 September 2017 / Published online: 10 October 2017

Abstract Feature selection methods have been used in various applications of machine learning, bioinformatics, pattern recognition and network traffic analysis. In high dimensional datasets, due to redundant features and curse of dimensionality, a learning method takes significant amount of time and performance of the model decreases. To overcome these problems, we use feature selection technique to select a subset of relevant and non-redundant features. But, most feature selection methods are unstable in nature, i.e., for different training datasets, a feature selection method selects different subsets of features that yields different classification accuracy. In this paper, we provide an ensemble feature selection method using feature–class and feature-feature mutual information to select an optimal subset of features by combining multiple subsets of features. The method is validated using four classifiers viz., decision trees, random forests, KNN and SVM on fourteen UCI, five gene expression and two network datasets.

Keywords Ensemble · Feature · Classification · Mutual-information

✉ Nazrul Hoque
tonazrul@gmail.com

Mihir Singh
mihir_singh23@yahoo.com

Dhruba K. Bhattacharyya
dkB@tezu.ernet.in

[1] Department of CSE, Kaziranga University, Jorhat, Assam, India

[2] Department of CSE, Tezpur University, Tezpur, Assam, India

Introduction

Feature selection is used to select a subset of relevant and non-redundant features from a large feature space. In many applications of machine learning and pattern recognition, feature selection is used to select an optimal feature subset to train the learning model. While dealing with large datasets, it is often the case that information available is somehow redundant for the learning process. The process of identifying and removing the irrelevant features from the original feature space, so that the learning algorithms can mainly focus on the relevant data which are useful for analysis and future predictions, is called feature or variable or attribute selection. The main objectives of feature selection are: (i) to improve predictive accuracy (ii) to remove redundant features and (iii) to reduce time consumption during analysis. Rodriguez et al. [23] state that performance of classification models improves when irrelevant and redundant features are eliminated from the original dataset. Moreover, a single feature selection method may generate local optimal or sub-optimal feature subset for which a learning method compromises its performance. In ensemble-based feature selection method, multiple feature subsets are combined to select an optimal subset of features using combination of feature ranking that improves classification accuracy. In the first step of ensemble method, a set of different feature selectors are chosen and each selector provides a sorted order of features. The second step aggregates the selected subsets of features using different aggregation techniques [28].

In the last two decades, a significant number of feature selection methods have been proposed that work differently using various metrics (like probability distribution, entropy, correlation etc) [1,3,11,12,15,22,26,29]. Feature selection methods are used to reduce dimensionality of data for big data analytics [8,16], gene expression data analysis and network traffic traffic analysis [7,13,19]. For a given dataset, different feature selection algorithms may select different subset of features and hence the result obtained may have different accuracy. So, people use ensemble-based feature selection method to select a stable feature set which improves classification accuracy. But, the main problem which needs to be considered in designing an ensemble-based feature selection is diversity [27]. Diversity may be achieved through using different datasets, feature subsets, or classifiers.

Types of feature selection

Filter approach [9] is used to select a subset of features from high dimensional datasets without using a learning algorithm. Filter-based feature selection methods are typically faster but the classifier accuracy is not ensured. Whereas, wrapper approach [4] uses a learning algorithm to evaluate the accuracy of a selected subset of features during classification. Wrapper methods can give high classification accuracy than filter method for particular classifiers but they are less cost effective. Embedded approach [9] performs feature selection during the process of training and is specific to the applied learning algorithms. Hybrid approach [14] is basically a combination of both filter and wrapper-based methods. Here, a filter approach selects a candidate feature set from the original feature set and the candidate feature set is refined by the wrapper approach. It exploits the advantages of both these approaches.

Discussion

As the dimensionality of the data is increasing day-by-day, difficulty in analyzing the data is also increasing with the same pace. In that context, feature selection is becoming an essential requirement in many data mining applications. From our empirical study, it has been observed that use of a single filter often fails to provide consistent performance on multiple datasets. So different filters can be ensembled to overcome the biasness or limitations of the identical classifiers and to provide consistent performance over a wide range of applications.

Motivation and problem definition

In literature, we found a large number of filter-based feature selection methods such as cfs, gain ratio, info gain and reliefF. The main problems of filter-based feature selection are (i) most of them do not consider the redundancy among the selected features, (ii) a single filter-based method may have biasness on selected feature subset and (iii) inconsistent prediction accuracy during classification. To overcome these problems, an ensemble of feature selection methods is introduced to select an optimal subset of non-redundant and relevant features which improves prediction accuracy during classification. However, most ensemble-based feature selection methods do not consider redundancy among the selected features during feature selection. For example, Canedo et al. [5] tested an ensemble-based feature selection method where the combiner simply takes the union of different subsets of features generated by multiple filter methods. After the experimental analysis on 5–15 datasets, they claim that the accuracy of their ensemble method was degraded. We are motivated to improve the classification accuracies of classifiers using an ensemble method that uses feature-class and feature-feature mutual information to combine different subsets of features. Feature-class mutual information is used to select relevant features whereas feature-feature mutual information is used to select non-redundant features. The union operation simply selects unique features from different subsets of features but it does not consider the redundant features in terms of prediction information or irrelevant features. Hence, we propose an ensemble feature-selection method that selects only a subset of relevant and non-redundant features. We formulate our problem as follows:

For a given dataset D, it is aimed to select initially different subsets of features, say $S_1, S_2, \ldots, S_n$ using n filter-based feature-selection methods. Next, is to combine these feature subsets using feature-class and feature-feature mutual information to generate an optimal feature subset, say F which contains only non-redundant, yet relevant features. The appropriateness of the feature subset is to be validated using unbiased classifiers on benchmark datasets.

Paper organization

The rest of the paper is organized as follows: In the section, “Related work”, we report related work in brief. In the section, “EFS-MI: the proposed MI-based ensemble feature selection”, we explain our proposed method and related concepts. The experimental results are analyzed in the section, “Experimental results” followed by conclusions and future work in the section, “Conclusion and future work”.

Related work

In the past two decades, a good number of ensemble feature selection methods have been proposed. Bagging [6] and boosting [25] are two most popular examples, which work on bootstrap samples of the training set. A bootstrap sample is a replica of dataset created by randomly selecting k instances, with replacement from the training set. Each of the replica is fed to a filter. Prediction of each classifier is combined using simple voting. On the other hand, the boosting approach samples the instances in proportion to their weights. An instance is weighed heavily if the previous model misclassified it. Olsson et al. [20] combines three commonly used ranker documents such as frequency threshold, information gain and chi-square for text classification problems. Wang et al. [28] present the ensemble of six commonly used filter-based rankers whereas Optiz [21] studies the ensemble feature selection for neural networks called genetic ensemble feature selection. Lee [18] and Rokachetal [24] combine outcomes of various non ranker filter-based feature subset selection techniques.

Moreover, feature selection methods have been applied in the classification problems such as bioinformatics and signal processing [33]. Generally, a cost-based feature selection method is used to maximize the classification performance and minimize the classification cost associated with the features, which is a multi-objective optimization problem. Zhang et al. [32] propose a cost-based feature selection method using multi-objective particle swarm optimization (PSO). The method generates a Pareto front of nondominated solutions, that is, feature subsets, to meet different requirements of decision-makers in real-world applications. In order to enhance the search capability of the proposed algorithm, a probability-based encoding technology and an effective hybrid operator, together with the ideas of the crowding distance, the external archive, and the Pareto domination relationship, are applied to PSO. A binary bare bones particle swarm optimization (BPSO) method is proposed to select an optimal subset of features [31]. In this method, a reinforced memory strategy is designed to update the local leaders of particles for avoiding the degradation of outstanding genes in the particles, and a uniform combination is proposed to balance the local exploitation and the global exploration of algorithm. The experiments show that the proposed algorithm is competitive in terms of both classification accuracy and computational performance. Canedo et al. [5] propose an ensemble of filters and classifiers for microarray data classification. Yu et al. [30] propose an ensemble based on GA wrapper feature selection, wherein they use three real-world data sets to show that the proposed method outperforms a single classifier employing all the features and a best individual classifier is obtained from GA based feature subset selection. Stephen Bay claims that ensemble of multiple classifiers is an effective technique for improving classification accuracy [2] and propose a combining algorithm for nearest neighbor classifiers using multiple feature subsets. Our study reveals that none of the existing ensemble methods are totally free from those limitations, as reported initially in section "Motivation and problem definition".

EFS-MI: the proposed MI based ensemble feature selection

To overcome the limitations, we introduced an ensemble feature selection method uses mutual information to select an optimal subset of features. In information theory, mutual information $I(X;Y)$ is the amount of uncertainty in X due to the knowledge of Y [17]. Mathematically, mutual information is defined as

$$I(X;Y) = \sum_{x,y} p(x,y) \log_2 \frac{p(x,y)}{p(x),\, p(y)} \tag{1}$$

where $P(x,y)$ is the joint probability distribution function of X and Y, and $P(x)$ and $P(y)$ are the marginal probability distribution functions for X and Y. We can also say

$$I(X;Y) = H(X) - H(X|Y) \tag{2}$$

where, $H(X)$ is the marginal entropy, $H(X|Y)$ is the conditional entropy, and $H(X;Y)$ is the joint entropy of X and Y. Here, if $H(X)$ represents the measure of uncertainty about a random variable, then $H(X|Y)$ measures what Y does not say about X. This is the amount of uncertainty in X after knowing Y and this substantiates the intuitive meaning of mutual information as the amount of information that knowing either variable provides about the other. In our method, a mutual information measure is used to calculate the information gain among features as well as between feature and class attributes. We define the marginal entropy and conditional entropy as follows.

Definition 1 *Marginal Entropy* For a random variable X, if the marginal distribution is $P(X)$ then the distribution has an associated marginal entropy which is defined as follows:

$$H(X) = \sum_{i} P(x_i) \log_2 \frac{1}{P(x_i)}.$$

Definition 2 *Conditional Entropy* If X and Y are two discrete random variables; $P(x,y)$ and $P(y|x)$ are joint and conditional probability distributions respectively, then the conditional entropy associated with these distributions is defined as

$$H(Y|X) = -\sum_{x \in X} \sum_{y \in Y} P(x,y) \log_2 P(y|x).$$

For better understanding let us take an example, assume a language with the following letters and associated probabilities as given in Table 1:

X	p	t	k	a	i	u
p(X)	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$

Table 1 Dataset result

DN	Dataset name	NoI	NoF	NoSF	SF (DRW)
1	Accut1	120	7	5	5,6,4,1,3
2	Accute2	120	5	5	1,4,7,5,2
3	Abalone	4177	7	4	3,5,6,1
4	Glass	214	9	5	3,8,7,5,6
5	Wine	178	13	5	7,10,1,2,4
6	Iris	150	4	3	3,1,2
7	Monk1	432	6	4	5,2,6,4
8	Monk2	432	6	3	5,2,1
9	Monk3	432	6	4	5,4,6,3
10	Tic-Tac Toe	958	9	5	5,3,9,1,7
11	Zoo	101	16	5	13,4,8,3,1
12	Diabetes	768	9	4	2,8,5,4
13	Car	1728	7	5	6,4,1,2,5
14	Heart-statlog	270	14	5	13,12,8,11,10
15	Breast-cancerW	699	10	5	2,3,7,1,5
16	Colon-cancer	62	2000	5	765,1771,1972,1672,737
17	Lung-cancer	73	325	5	23,243,27,205,12
18	Lymphoma	45	4026	5	734,760,37,853,2659
19	SRBCT	83	2308	5	509,107,1105,1915,1916
20	TUIDS		20	5	17,16,19,18,6
21	NSL-KDD		43	5	5,3,4,6,42

NoI number of instances, *NoF* number of features, *NoSF* number of selected features, *SF (DRW)* selected features (decreasing rank wise)

Here, marginal entropy of X is:

$$H(X) = -\sum_{(p,t,k,a,i,u)} P(x)\log_2 P(x)$$

$$H(X) = -\left(4\times(1/8)\times\log_2\frac{1}{8}+2\times(1/4)\times\log_2\frac{1}{4}\right)=2.5$$

The joint probability of a vowel and a consonant occurring together is given in Table 2:

$P(x,y)$	p	t	k	$P(y)$
a	$\frac{1}{16}$	$\frac{3}{8}$	$\frac{1}{16}$	$\frac{1}{2}$
i	$\frac{1}{16}$	$\frac{3}{16}$	0	$\frac{1}{4}$
u	0	$\frac{3}{16}$	$\frac{1}{16}$	$\frac{1}{4}$
$P(x)$	$\frac{1}{8}$	$\frac{3}{4}$	$\frac{1}{8}$	

Now, the conditional entropy of a vowel and a consonant can be computed as follows:

$$H(V|C) = -\sum_{x\in X}\sum_{y\in Y} P(x,y)\log_2 P(y|x)$$

$$\begin{aligned}H(V|C) = &-(P(a,p)\log_2 P(a|p)+P(a,t)\log_2 P(a|t)\\&+P(a,k)\log_2 P(a|k)+P(i,p)\log_2 P(i|p)\\&+P(i,t)\log_2 P(i|t)+P(i,k)\log_2 P(i|k)\\&+P(u,p)\log_2 P(u|p)+P(u,t)\log_2 P(u|t)\\&+P(u,k)\log_2 P(u|k))\end{aligned}$$

$$\begin{aligned}H(V|C) = &-\left(\frac{1}{16}\log_2\frac{\frac{1}{16}}{\frac{1}{8}}+\frac{3}{8}\log_2\frac{\frac{3}{8}}{\frac{3}{4}}\right.\\&+\frac{1}{16}\log_2\frac{\frac{1}{16}}{\frac{1}{8}}+\frac{1}{16}\log_2\frac{\frac{1}{16}}{\frac{1}{8}}+\frac{3}{16}\log_2\frac{\frac{3}{16}}{\frac{3}{4}}+0\\&\left.+0+\frac{3}{16}\log_2\frac{\frac{3}{16}}{\frac{3}{4}}+\frac{1}{16}\log_2\frac{\frac{1}{16}}{\frac{1}{8}}\right)\\&=\frac{11}{8}=1.375\end{aligned}$$

EFS-MI: Algorithm

Our proposal is an ensemble approach that combines the subsets obtained from various filters using feature-class and feature-feature mutual information as shown in Fig. 1. The method combines the subsets of features selected by different feature selection methods using greedy search approach. For a particular rank, if a common feature is chosen by all the selectors, then that feature is selected without using greedy search technique and put into the optimal subset.

Table 2 Classification accuracies and average classification accuracies (ACA) for network intrusion datasets

D	Decision tree							Random forest						KNN					
	Rank	SU	GR	IG	RF	CS	EFS	SU	GR	IG	RF	CS	EFS	SU	GR	IG	RF	CS	EFS
20	$f = 5$	0.95	0.90	0.95	0.93	0.95	0.95	0.96	0.89	0.96	0.88	0.96	0.92	0.96	0.87	0.95	0.9	0.96	0.97
	$f = 3$	0.96	0.95	0.95	0.9	0.95	0.96	0.97	0.95	0.97	0.95	0.97	0.93	0.93	0.96	0.92	0.87	0.96	0.96
	$f = 4$	0.96	0.97	0.95	0.94	0.96	0.96	0.98	0.97	0.98	0.97	0.97	0.96	0.92	0.96	0.93	0.88	0.93	0.96
	$f = 6$	0.97	0.97	0.97	0.95	0.97	0.97	0.98	0.98	0.98	0.97	0.98	0.97	0.92	0.94	0.93	0.88	0.92	0.95
	$f = 42$	0.97	0.97	0.97	0.95	0.97	0.97	0.98	0.98	0.98	0.97	0.98	0.98	0.92	0.93	0.93	0.88	0.93	0.94
	$f = 33$	0.97	0.97	0.97	0.95	0.97	0.97	0.98	0.98	0.98	0.97	0.98	0.98	0.92	0.92	0.92	0.90	0.92	0.95
	$f = 23$	0.97	0.97	0.97	0.96	0.97	0.97	0.98	0.98	0.98	0.97	0.98	0.98	0.92	0.92	0.92	0.91	0.92	0.96
	$f = 34$	0.97	0.97	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.92	0.92	0.92	0.92	0.92	0.96
	ACA	0.95	0.92	0.95	0.94	0.97	0.96	0.97	0.94	0.96	0.91	0.96	0.96	0.92	0.91	0.95	0.88	0.96	0.95
21	$f = 1$	0.79	0.98	0.98	0.98	0.98	0.99	0.79	0.98	0.98	0.98	0.98	0.98	0.79	0.98	0.98	0.98	0.98	0.98
	$f = 17$	0.99	0.99	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.99
	$f = 16$	0.99	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.98	0.98	0.98	0.98	0.99
	$f = 19$	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.98	0.98	0.98	0.98	0.99
	$f = 18$	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.98	0.98	0.98	0.98	0.99
	ACA	0.95	0.98	0.98	0.98	0.98	0.99	0.95	0.98	0.98	0.98	0.98	0.98	0.94	0.98	0.98	0.98	0.98	0.98

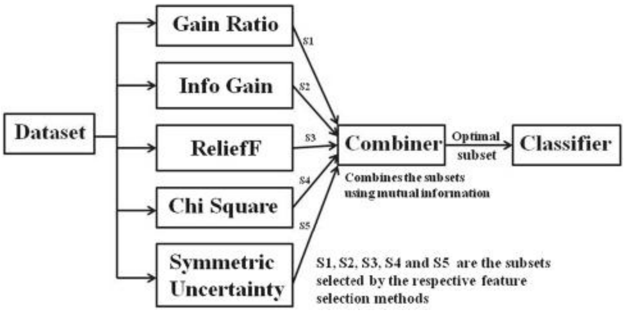
Gain Ratio
Info Gain
Dataset
ReliefF
Chi Square
Symmetric Uncertainty
S1
S2
S3
S4
S5
Combiner
Optimal subset
Classifier
Combines the subsets using mutual information
S1, S2, S3, S4 and S5 are the subsets selected by the respective feature selection methods

Fig. 1 Proposed framework

Otherwise, the method computes feature–class and feature-feature mutual information and selects a feature that has maximum feature–class mutual information but minimum feature-feature mutual information. It removes redundancy among the selected features using feature-feature mutual information and selects relevant features using the feature–class mutual information and hence removes the biasness induced by the individual feature-selection methods. The steps of the algorithm are shown in Algorithm 1.

Function of a combiner

The combiner combines the selected subset of features based on feature–class and feature-feature mutual information. First, the combiner considers the first raked features from all the selected subsets and if all the first-ranked features are same then without computing feature–class and feature-feature mutual information, we pick up that common feature as an optimal feature. But, if the features are different then first we compute feature–class mutual information for every feature and consider the feature that has the highest feature–class mutual information. Now, for this feature we compute feature-feature mutual information with all the features that are already been selected as optimal features and if the feature-feature mutual information of that feature with all other selected features is less than a user defined threshold say, α then the feature will be selected. The feature-feature mutual information is used to measure the feature-relevance of a non-selected feature with selected features. We introduce an effective value for α ($\alpha = 0.75$) based on our exhaustive experimental study. In ensemble approach, a ’combiner’ plays an important role in ensembling various feature selection methods. People use different methods such as majority voting, weighted voting, sum rule, mean rule, product rule, maximum rule, minimum rule, correlation and mutual information to build the combiner. However, the combiner for the proposed ensemble feature selection method emphasizes on reducing the redundancy among the selected subset of features by incorporating feature–class as well as feature-feature mutual information. To explain our method, following definitions are useful.

Definition 3 Feature–class relevance: It is defined as the degree of feature–class mutual information between a feature f and a class label C.

Definition 4 Feature-feature relevance: It is defined as the degree of feature-feature mutual information between any two features, say f_i and f_j where $f_i, f_j \in F$.

Definition 5 Redundant feature: Two non-selected features say f_i and f_j are defined as redundant with respect to a given selected feature say f_k, $if\ feature\_feature\_MI(f_i, f_k) = feature\_feature\_MI(f_j, f_k)$.

Lemma The subset of features identified by EFS-MI is non-redundant and relevant.

Proof The proposed EFS-MI selects a feature f_i as relevant only when the feature–class relevance between f_i and a given class, say C is high (as per Definition 1). Again, for a given selected feature f_k, EFS-MI considers any two features say f_i and f_j as redundant, if $feature\_feature\_MI(f_i, f_k) = feature\_feature\_MI(f_j, f_k)$ and exclude from further consideration. Hence the proof. □

A feature $f_i \in F$ identified by EFS-MI, if f_i has a high feature–class relevance based on mutual information with the class label C (as per Definition 1). Further, for any two given features, f_j and f_k, our method EFS-MI shall include only f_i in the final subset of features if the feature-feature mutual information of f_i is smaller than both f_j and f_k , i.e., shall discard both the redundant features f_j and f_k (as per Definition 3).

Lemma EFS-MI removes biasness of individual feature selection method during ensemble.

Proof The proposed method considers the following two situations to remove biasness of each individual feature selection method.

(i) if a feature f_i is selected by all the individual feature selection methods say M_j, for rank R_k, the feature is intuitively placed in the optimal subset at rank R_k.
(ii) if the features f_i is not same for all the individual feature selection methods M_j, for a particular rank R_k, the biasness of a feature selection methods is removed using $feature\_feature\_MI()$ and $feature\_class\_MI()$. EFS-MI selects a feature f_i if its feature-class mutual information is comparatively high but the feature-feature mutual information with all other features selected already are comapratively low. □

In algorithm 1, the value of K is an user input that represents the number of selected features in the optimal set. The value of K is decided heuristically based on empirical study. The proposed method first considers the highest ranked feature and computes the classification accuracy using a classifier. In the next iteration, it considers a subset of features that includes both the first-ranked feature as well as the second-ranked feature, and then the first-, second- and third-ranked features and so on. For each subset of features, the method computes classification accuracy. If a subset of features that includes K high ranked features gives the highest classification accuracy compared to another subset that includes K+1 high ranked features, then the subset of K high ranked features is considered as an optimal subset of features. From the empirical analysis, we found that the number of selected features is in the range 3–5.

```
Data: D is the number of datasets, K is the number of features to
      be included in the optimal set, $\alpha$ is a user defined threshold
Result: $F$, an optimal subset of features
Steps:
Select the subsets $S_1, S_2, \cdots, S_n$ using different filter-based
methods
Initialize $F \leftarrow \phi$ and a counter $j \leftarrow 1$
while $j \leq K$ do
    if $(\{S_1[j]\} = \{S_2[j]\} = \{S_3[j]\} \cdots, \{S_{n-1}[j]\} =$
    $\{S_n[j]\})$ then
        $F \leftarrow F \cup \{S_1[j]\}$
    else
        Compute feature-class_MI $(\{S_i[j]\}, C)$ ,
        $\forall i \in [1, n]$
    end
    Select the feature $f$ with maximum feature-class mutual
    information
    if (F==NULL) then
        $F \leftarrow F \cup \{f\}$
    else
        Compute feature-feature_MI(f, $f_s$), for $f$ with all the
        selected feature $f_s \in F$
        if Calculated information is less than $\alpha$ for all selected
        features in $F$ then
            $F \leftarrow F \cup \{f\}$
        end
    end
     j←j+1;
end
```

Algorithm 1: EFS-MI

Complexity analysis

Let size of each feature subset is K and the number of subsets obtained is n, then for the outer loop complexity is $O(K \times n)$. If at any time total number of features included in F is m, the function $feature\_feature\_MI()$ loop iterates for $O(m)$ times. Thus, the overall complexity of the above algorithm is $O(K \times n) + O(m)$ approximately. If K is very large then complexity will be almost linear because the number of filters employed will be less as compared to the value of K. If the value of K and m are also very large, overall complexity will be linear.

Experimental results

The proposed EFS-MI feature selection method is implemented in MATLAB 2008 software. We carried out the experiment on a workstation having 12 GB main memory, 2.26 Intel(R) Xeon processor and 64-bit Windows 7 operating system. Also, we use a freely available toolbox called Weka [10] where many feature-selection algorithms are available.