AN ENSEMBLE OF FILTERS AND WRAPPERS FOR MICROARRAY DATA CLASSIFICATION

Mohamad Morovvat[1] and Alireza Osareh[2]

[1]MS Holder of Artificial Intelligence, Department of Computer Engineering,
Shahid Chamran University of Ahvaz, Ahvaz, Iran

[2] Associate Professor of Computer Engineering, Department of Computer Engineering,
Shahid Chamran University of Ahvaz, Ahvaz, Iran

ABSTRACT

The development of microarray technology has supplied a large volume of data to many fields. The gene microarray analysis and classification have demonstrated an effective way for the effective diagnosis of diseases and cancers. In as much as the data achieving from microarray technology is very noisy and also has thousands of features, feature selection plays an important role in removing irrelevant and redundant features and also reducing computational complexity. There are two important approaches for gene selection in microarray data analysis, the filters and the wrappers. To select a concise subset of informative genes, we introduce a hybrid feature selection which combines two approaches. The fact of the matter is that candidate's features are first selected from the original set via several effective filters. The candidate feature set is further refined by more accurate wrappers. Thus, we can take advantage of both the filters and wrappers. Experimental results based on 11 microarray datasets show that our mechanism can be effected with a smaller feature set. Moreover, these feature subsets can be obtained in a reasonable time.

KEYWORDS

Ensemble, Feature selection, Filters, Wrappers, Microarray Data.

1. INTRODUCTION

Microarray technology has provided the ability to measure the expression level of thousands of gene simultaneously in a single experiment. With a certain number of samples, investigations can be made into whether there are patterns or dissimilarities across samples of different type including cancerous versus normal, or even within subtype of diseases [1].

Microarray analysis has been challenged by its high number of features (genes) and the small sample sizes (for example lung dataset [2] contains 12535 genes and only 181 samples). Therefore, feature selection and classification are two essential steps in order to predict a person's risk of cancer.

To avoid the curse of dimensionality problem, gene selection plays a crucial role in DNA microarray analysis. Another important reason to reduce dimensionality is to help biologists to identify the underlying mechanism that relates gene expression to diseases.

The development of feature selection is divided into two major directions. One is the filters and the other is the wrappers. In the filters approach, a good feature set is selected as a result of

DOI:10.5121/mlaij.2016.3201

preprocessing based on properties of the data itself and independent of the classification algorithm. In this paper, methods such as Correlation-based Filter Selection (CFS), FCBF, GSNR, ReliefF, minimum Redundancy Maximum Relevance (mRMR) have been shown to be effective scores for measuring the discriminative power of gens.

In fact, although gene selection using filters are simple and fast, the method suffers from several major drawbacks:

- In some filters approach correlation between genes is not taken into account.
- The filters work fast using a simple measurement, but its result is not always satisfactory.

On the other hand, the wrappers guarantee good results through examining learning result, but they are so slow when the feature set is wide. In the wrapper approach, genes are selected sequentially one by one so as to optimize the training accuracy of a particular classifier [3] that is, the classifier is first trained using one single gene, and this training is performed for the entire original gene set. The gene that gives the highest training accuracy is selected. Then, a second gene is added to the selected gene and the gene that gives the highest training accuracy for the two-gene classifier is chosen. This process is continued until a sufficiently high accuracy is achieved with a certain gene subset.

In as much as both of filters and wrappers have its own advantages and disadvantages, in this paper a hybrid method is proposed that is combined with both of them called an ensemble of filters and wrappers. In fact, in our approach the filters select several important features and then, the wrapper is applied in order to optimize classification accuracy in final gene selection. Thus, we can achieve a good subset of features.

To this end, three different types of classifications are selected namely J48, SMO and Naïve bayes in order to classify each sample on each one of datasets. In fact, combining the output of several classifiers may reduce the risk of selecting a poorly performing classifier. In so doing, after applying these classifiers, we propose a combination method and use the power of all classifiers in order to improve the accuracy.

1.1. Existing Work

In recent years, researchers have shown an increased interest in microarray data classification and different methods are proposed by researchers in order to achieve a good accuracy. A considerable amount of literature has been published on microarray data classification. In this section, we mention several existing works.

So far, many machine learning algorithms have been introduced and many of them have been employed for both steps, including the techniques of feature selection [4], and classification techniques, e.g. K-NN [5], support vector machines [6, 7] and neural networks [8]. Most of the existing research works attempt to choose an optimal subset of genes and then generalize an accurate classification model based on the selected genes.

Many methods have been proposed in microarray classification, including subspace clustering, for example Song et al [9] presented a novel clustering-based feature subset selection algorithm for high dimensional data.

Ensemble methods are another method which has attracted the attention of researches and researchers have used many terms to describe the combining models involving different learning algorithms. Canedo et al [10] described a new framework for feature selection consisting of an ensemble of filters and classifiers. And also Nagi et al [11] combined the results of Boosting, Bagging and Stacking to obtain results which are significantly better than using Boosting, Bagging or Stacking alone. Liu et al [12] proposed a new ensemble gene selection method in which each gene subset is obtained by the same gene selector with different starting point.
Recently, another thing that has attracted the researchers attraction is related to evolutionary algorithms. For example Hala et al [13] proposed a new hybrid gene selection namely Genetic Bee Colony(GBE). In this paper both Genetic Algorithm and Artificial Bee Colony have been applied to select the most informative and predictive genes for microarray classification.

1.2. Organized of the Paper

The rest of this paper is organized as follows: Section 2 provides the background of different stages of the work. The hybrid method is described in section 3. In Section 4, the experimental results and corresponding discussions are presented. Section 5 concludes the paper.

2. MATERIALS AND METHODS

2.1. Preprocessing Data

In order to achieve more accurate results, data pre-processing is an important step for handling gene expression data. This includes two steps: filling missing values and normalization. Different type of methods for dealing with these two steps are available. In our paper for both training and test dataset, missing values are filled using the average value of that gene. Normalization is then carried out so that every observed gene expression has mean equal to 0 and variance equal to 1.

2.2. Gene Selection

Feature selection methods have been applied to classification problem so as to select a reduced a feature set that makes the classifier more accurate and faster. There are two broad categories for feature selection algorithms, filter model and wrapper. In DNA microarray data, this fact that the ratio between the number of samples and the number of features is very small, prevents the use of a wrapper model at first because it could not be generalized adequately. Therefore, at the first stage, we choose several popular filter methods with different metrics in order to achieve a good substance of features. Every filter has its own characteristics and when we use several filters instead of one filter, we can take advantage of different approaches and this issue can guarantee to achieve more good features. The filters that have been chosen as a filter approach in our paper, have been described in below:

- CFS: This is a simple filter algorithm that ranks feature subsets according to a correlation-based heuristic evaluation function [14].
- FCBF: In the first step, a subset of relevant features whose C-correlation are larger than a given threshold are selected, and then sorts the relevant features in descending order in term of C-correlation [15]. Finally, redundant features are eliminated one-by-one in a descending order.

- Symmetric Uncertainty (SU): SU is an extension of the information gain. The fact of the matter is that, Symmetrical uncertainty compensates for information gain's bias toward attributes with more values and normalizes its value to the range [0, 1] [16]
- ReliefF: ReliefF [17] is an extension of the original Relief algorithm [18] that adds the ability of dealing with multiclass problems and it is more robust and capable of dealing with incomplete and noisy data.
- mRMR: The mRMR criterion computes both the redundancy between features and the relevance of each feature [19].
- GSNR: It has been proposed and used in [20]. The GSNR is a measure of the ratio between inter-group and intra-group variations. Higher GSNR values indicate higher discrimination power for the gene.

Two points are important to be concerned about those features that are described above:

- The first two provide a subset of features, whereas the last four provide features ordered according to the irrelevance (a ranking of features).
- Some filters don't take into account correlation between genes and they just evaluate correlation between features and target concept, like SU filter. These types of filters are not successful in removing redundant features which are not irrelevant but are covered by other features and result in increase in the complexity of problem.

In general, as it is explained in [21], we can consider different types of features in four categories: irrelevant features (I), redundant features (II, part of weakly relevant features), weakly relevant but non-redundant features (III), and strongly relevant features (IV). As it is shown in figure 1, an optimal subset essentially contains all the features in parts III and IV.

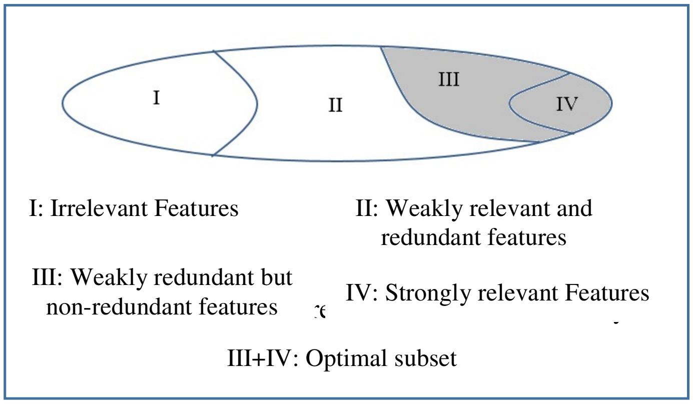
I
II
III
IV
I: Irrelevant Features
II: Weakly relevant and redundant features
III: Weakly redundant but non-redundant features
IV: Strongly relevant Features
III+IV: Optimal subset

Figure 1. A view of feature relevance and redundancy.

Irrelevant features, along with redundant features, severely affect the accuracy of the learning machines, and therefore in order to achieve a good subset of features, we are trying to remove these types of features. Although some feature selections can be used to remove both irrelevant and redundant features (like FCBF), it is better to apply a specific filter in order to improve time complexity to find a subset of features. Thus, in our paper the SU is selected to eliminate irrelevant features at first. After that, other filters are applied so as to improve the quality of subsets of features. When the number of features was decreased, the wrappers are applied to improve the accuracy.

2.3. Classifiers

There are different base learning algorithms, meantime we select three classifiers i.e. j48(Decision tree), Naïve Bayes(Probabilistic) and SMO(Sequential minimal optimization -SVM) based on the following ground:

- According to previous study [10, 22] All these classifiers performed consistently well in microarray data.
- They are from three different classifications of algorithm.
- They belong to three different statements i.e. unstable, probabilistic and stable.

Every classifier has its own advantages and disadvantages and when you select different types of classifiers, you can take advantage of all of them. Of course, it depends on the way you select in order to combine the result

2.4. Datasets

Here, we utilized 11 publicly available benchmark datasets [15]. A brief overview of these datasets is summarized in Table 1. As it can be seen in table 1, the number of features is so high, whereas the number of sample is so low in all dataset. This is exactly the challenge that microarray data are involved.

Table 1. Summary of bench-mark gene microarray datasets.

Dataset	**# Total Genes (T)**	**# Instances (n)**	**# Classes (C)**
Colon Tumor	2000	62	2
Central Nervous System	7129	60	2
Leukaemia	7129	72	2
Breast Cancer	24481	97	2
Ovarian Cancer	15154	253	2
MLL	12582	72	3
Lymphoma	4026	66	3
Leukaemia-3C	7129	72	3
Leukaemia-4C	7129	72	4
SRBCT	2308	83	4
Lung Cancer	12600	203	5

3. AN ENSEMBLE OF FILTERS AND WRAPPERS

In this section we describe our hybrid feature selection procedure. Our approach is composed of four sections i.e. choosing several filters in order to eliminate irrelevant and redundant feature, applying wrappers so as to achieve a good accuracy, applying classifiers and combining result in order to make a decision. An architecture of our model is indicated in figure 2.

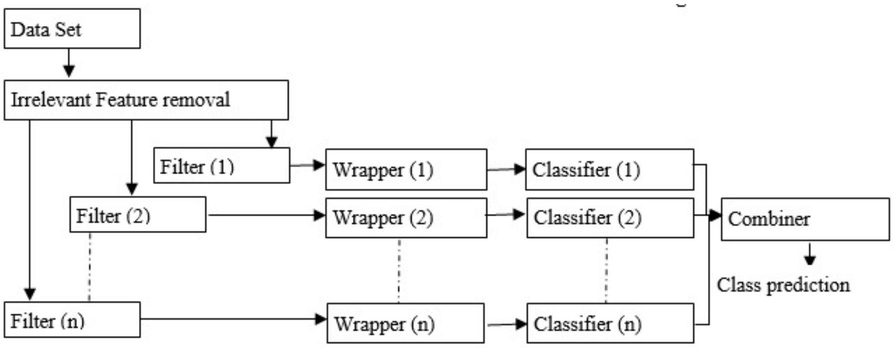

Figure 2. Architecture of our proposed model.

3.1. Removal Irrelevant Features

In as much as irrelevant features don't have any contribution to the predictive accuracy, it is better to be removed in order to improve time complexity. Thus, we have taken advantages of SU (an entropy-based filter method) in order to remove these such features. The SU is derived from the mutual information by normalizing it to the entropies of feature values or feature values and target classes [9]. Therefore, we choose SU as the measure of correlation between the feature and the target concept, and also has been used to evaluate the goodness of features for classification by a number of researchers (e.g., Hall [14], Hall and Smith [23], Yu and Liu [24], [25], Zhao and Liu [26], [27]).

The symmetric uncertainly is defined as follows:

$$SU(x, y) = \frac{2 * Gain(x \mid y)}{H(x) + H(y)}$$

Where,

- H(x), H(y) are the entropy of discrete random variable X,Y Respectively.
- $Gain(x \mid y)$ is the amount by which the entropy of Y decreases. It reflects the additional information about Y provided by X and is called the information gain [28].

The fact of the matter is that SU treats a pair of variables symmetrically and it compensates for information gain's bias toward variables with more values and normalizes its value to the range [0,1]. A value 1 of (X,Y) indicates that knowledge of the value of either one completely predicts the value of the other and the value 0 reveals that X and Y are independent.

After calculating the SU relation for all features, features that have little relevance to the target set (irrelevant feature) should be removed. Thus, after sorting the features according to their SU values, we select till the $\left\lfloor \sqrt{m} * \log^{m} \right\rfloor$ *th* ranked feature for each datasets. It should be mentioned that this value is set heuristically [19].

3.2. Choose several filter in order to eliminate redundant feature

Irrelevant features along with redundant feature have effect on the accuracy of the learning algorithm. Thus, a good subset of features is a subset that lacks irrelevant and redundant features. In other words, a good subset of features contains some features highly correlated with class, yet uncorrelated with each other. Thus, after removing irrelevant features by SU, in this stage we select five popular feature selection, namely CFS, FCBF, ReliefF, mRMR, GSNR that were described in section 2.2. It has to be noted that, the first two provided a subset of features, whereas the last three provided features ordered according to their relevance (a ranking of features). For the second type, like SU filter, we have selected till the $\left\lfloor \sqrt{m} * \log^{m} \right\rfloor_{th}$ ranked feature for each datasets [9].

After applying each one of these methods on each one of datasets involved in this work and achieving some feature subsets, in order to determine the effectiveness of each filter, we have calculated the accuracy of each learning algorithm (J48, SMO, Naïve bayes) on each datasets according to those feature subsets that have been obtained by each filter. Finally, the best filters on a specific classifier are selected.

3.3. Apply wrappers so as to achieve a good accuracy

In as much as the wrapper is very slow when applied to wide feature sets which contain hundreds or even thousands of features (like microarray data), it is not affordable to apply at first because of the computational time and complexity. But after applying the filters and achieving a good subset, it is useful to use wrapper in order to achieve good result through examining learning results and consequently we can take advantage of the simplicity of the filter approach for initial gene screening and then make use of the wrapper approach to optimize classification accuracy in final gene selection.

3.4. Apply classifiers and combine result in order to make a decision

After achieving a good substance of features by filters and wrappers, we need to apply classifiers on each one of datasets in order to evaluate the power of our proposed hybrid feature selection. Something that has been the focus of much attention in recent years is ensemble of classification instead of one base classifier. The idea builds on the assumption that combining the output of multiple expertise is better than the output of any single expert. Thus, we use three different classifier namely J48, SMO, Naïve bayes. After applying several classifiers on each dataset, one of the most important things is the way that is selected for combining the result. Majority voting is one of the oldest strategies for decision making [29]. Three consensus patterns, unanimity, simple majority, and plurality are illustrated in Figure 3. If we assume that black, gray and white correspond to class labels, and the decision makers are the individual classifiers in the ensemble, the final label will be "black" for all three patterns.

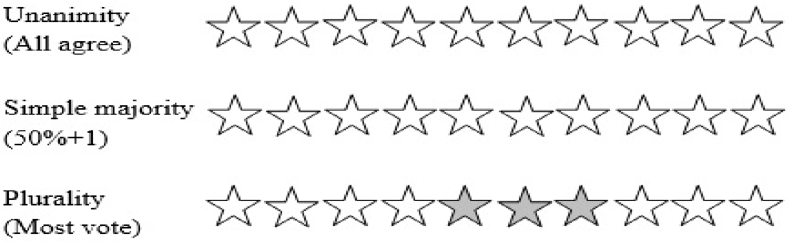

Figure 3. Consensus patterns in a group of 10 decision makers: unanimity, simple majority, and plurality. In all three cases the final decision of the group is "black."[30]

One of the most frequently used rule from the majority vote group is plurality and we also use it in order to make a decision. With respect to number of classifiers (3 classifiers), we don't have any problem for those datasets which contain 2 classes. But the situation is different for those datasets which contain more than 2 classes, because we have three ideas that are driven by three classifiers (e.g. lung cancer that has 5 classes and maybe each classifier chooses different class for one specific sample).

For this problem we use the level of expertise of each one of classifier. This means that, the more expertise a classifier has on a specific class, the more effect it has on that class.

Initially, the base classifiers are trained with the distinct training dataset. Next, we evaluate the performance of the base classifiers using the test dataset. Then, the classifier with the highest class performance for a certain class out of the base classifiers becomes the expert of that class. The class specific performance of a classifier is calculated as: [11]

Class specific accuracy= (Total no. of correctly predicted instances for a class)/ (total no. of predicted instances of that class).

We use the confuse matrix in order to calculate the class performance of a classifier. As an example, if we consider table 2 as a confusion matrix for j48 classifier on MLL dataset, then the class-specific accuracy for class B for J48 classifier is 0.85(17/(17+3)). It means that J48 classifier is expert as much as 0.85 on b class in MLL dataset. This way, we compute this factor for each base classifier on each class and store those in a matrix.

Table 2. Confusion matrix for j48 classifier on MLL dataset.

	a	**b**	**C**
a=ALL	24	0	0
b=MLL	0	17	3
c=AML	0	1	27

During classification of an instance, the instance is first classified by the base classifiers and the individual predictions of the base classifiers are combined as follows:

- For one specific instance, if all the classifiers predict the same class, the result for that specific instance is the same class.
- If the predictions of majority classifiers (2 of 3) match, the result for that specific instance is the opinion which is related to majority of classifiers.
- If the predictions of all the classifiers disagree, then any of the following situations may arise:

  - ✓ One of the classifier is more expert than other in its prediction (that class which predicts by that specific classifier), then the ensemble goes by that classifier's decision.

- ✓ Two of the classifiers are equally expert in its prediction and also are more expert than other one, then we have an overall view to confuse matrix related to those two of the classifiers and the decision of the classifier which has a higher accuracy is taken as the final decision.(For example in tabel2, accuracy in overall view for J48 on MLL database is equal to $\frac{(24/(24+0+0))+(17/(0+17+3))+(27/(0+1+27))}{3}$=0.939)
- ✓ All the three classifiers could be the same expert in its class predictions. In that case, we have operated like (b) situation.

In this manner, the result of base classifiers is combined for a specific instance and we can overcome the problem that was related to those datasets having more than 2 classes.

3.5 Algorithms for proposed model

The detail of our proposed method is collected in 4 algorithm that has been shown in below. Algorithm 1 explains how to remove irrelevant features. In order to achieve three of the best subsets for each dataset, algorithm 2 is embedded. After achieving the set subset of features for each dataset, algorithm 3 is employed so as to apply wrapper for improving subset. Finally, after having good subset, it is enough that, algorithm 4 is applied in order to classify instance.

Algorithm 1. Irrelevant features removal.

```
Input: Microarray datasets
Output: First set of features for each dataset
For each microarray dataset do
    For each gene in a specific dataset do
        Calculate the SU relation
    Sort them in a descending order
    select till the $\lfloor \sqrt{m} * \log^{m} \rfloor$ $th$ ranked feature as a first set of feature for that especial dataset
```

Algorithm 2. Select three of the best subsets for each dataset.

```
Input: First set of features for each dataset
Output: three of the best subsets for each data set by filters
For GSNR, mRMR, ReliefF do
    For First set of features for each dataset do
        For each gene in a specific dataset do
            Calculate the statistical scores by that special filter
        Rank the scores from the highest to the lowest
        Select till the $\lfloor \sqrt{m} * \log^{m} \rfloor$ $th$ ranked feature as a second set of features for each dataset by that
special filter
Else //==for CFS,FCBF==//
    For First set of features for each dataset do
        Obtain subset of feature as a second set of features for each dataset by that special filter
For each learner do
    For each set of subset that has obtained by each filter  do  //==there are five sets of feature==//
        For second set of features for each dataset do
            Calculate the accuracy by that special learner
        Calculate the average of accuracy on that special set
    Rank the average of accuracy from the highest to the lowest
    Select the first set of subset for that learner
```

Algorithm 3. Apply Wrappers.

```
Input: three of the best subsets for each data set by filters
Output: three of the best subsets for each data set by wrapper
For each datasets do
    For each three of the best subsets by filters do
```

```
(3)     Initialize P=null(the set of selected gene) and n=0 (number of gene in P)
(4)     Put n=n+1;
(5)     repeat for all gene in that special best subset of features(F) $g_n = F\setminus P$
(6)         Using P ∪ {$g_n$} as the gene set, classify all samples using an specific classifier
(7)         Based on unanimous voting, determine the number of samples that are classified correctly.
(8)     Select the gene subset with the highest number of sample that is classified correctly.
(9)     Update p by adding the selected gene to the set.
(10)    Repeat number 4-8 until cannot improve the accuracy of learning algorithm.
(11)    Consider P as a one of three of the best subsets for that special dataset
```

Algorithm 4. Apply classifiers and combine result.

```
Input: three of the best subsets for one data set by wrapper
Output: classify each sample
(1) For all instance(ins) do
(2)     TestData=ins;
(3)     TrainingData= all instances-TestData;
(4)     For each learner in a specific subset do //==every of three subset belongs to specific classifier==//
(5)         Classifier= learner(TrainingData)
(6)         Predict by an specific learner=Apply classifier to TestData;
(7)         IF all classifiers predict the same class then the result for that specific instance is the same class.
(8)         Else IF the predictions of majority classifiers (2 of 3) match then the result for that specific instance is
        the opinion which is related to majority classifiers.
(9)         ELSE
(10)            Calculate the class specific performance of each classifier
(11)            IF One of the classifier is more expert than other in its prediction, Then the ensemble goes by that
        classifier's decision.
(12)            Else IF (Two of the classifier are equally expert in its prediction and also are more expert than
        other one) OR (All the three classifiers could be the same expert in its class predictions) Then
(13)                Calculate the accuracy for each learner
(14)                The classifier which has a higher accuracy is taken as the final decision.
```

4. EXPERIMENTAL RESULTS

To evaluate the performance of our proposed method and compare it with other approaches which have been done by other researchers, we bring the result of our experimental in this section. According to our approach, after filling missing values and normalization for each dataset, we have applied the SU in order to reduce the computational complexity of the problem at hand and also remove the irrelevant features. Table 3 shows the number of features that is selected by SU filter. As it can be seen in table 3, the number of features is so lower than the number of initial features. It should be noted that, after applying the SU filter on each dataset and sort them in a descending order according to amount of them, we select till the $\left\lfloor \sqrt{m} * \log^{m} \right\rfloor th$ ranked feature for each datasets that is set heuristically.

Table 3. Number of selected genes before/after applying SU filter.

Dataset	# Total Genes (T)	# Gene After SU
Colon Tumor	2000	147
Central Nervous System	7129	325
Leukaemia	7129	325
Breast Cancer	24481	686
Ovarian Cancer	15154	514
MLL	12582	459

Lymphoma	4026	228
Leukaemia-3C	7129	325
Leukaemia-4C	7129	325
SRBCT	2308	162
Lung Cancer	12600	460

After applying the SU filter, in order to achieve a good substance of feature, we run all 5 feature selection algorithms against each dataset and obtain the number of selected features for each algorithm. Table 4 shows the number of genes which are selected by these feature selection algorithms for each individual microarray gene dataset. As it can be seen, the number of selected genes for each processed gene dataset is different and it depends on the choice of a feature selection algorithm.

Table 4. Number of selected genes for each gene selection algorithm.

Dataset	**# Total Genes (T)**	**CFS**	**FCBF**	**GSNR**	**ReliefF**	**mRMR**
Colon Tumor	2000	26	14	26	26	26
Central Nervous System	7129	39	29	45	45	45
Leukaemia	7129	56	43	45	45	45
Breast Cancer	24481	136	87	74	74	74
Ovarian Cancer	15154	25	17	61	61	61
MLL	12582	64	50	57	57	57
Lymphoma	4026	56	50	35	35	35
Leukaemia-3C	7129	72	40	45	45	45
Leukaemia-4C	7129	71	46	45	45	45
SRBCT	2308	73	64	28	28	28
Lung Cancer	12600	92	68	57	57	57

As it was noted in previous part, GSNR, ReliefF and mRMR algorithms provide an ordered list of the initial genes (features) according to the genes importance and discrimination power. Here, for the sake of comparison, we experimentally retained till the $\left\lfloor \sqrt{m} * \log^{m} \right\rfloor th$ ranked feature by each of these three feature selection algorithm. This in turn leads to less computational cost in experiments.

To evaluate the gene classification accuracy of selected top genes by each feature selection algorithm, three learning algorithms are utilized. The learning algorithms apply on each newly obtained dataset containing only the selected genes, and in each case the final overall accuracy is measured.

Table 5 and table 6 summarize the learning accuracy of three classifiers on different feature sets. Considering the averaged accuracy over all datasets as have been summarized in table 7.

Table 5. Classification results obtained by three learning algorithm against different subset that have obtained by CFS and FCBF filters.

Dataset	**Filter name= CFS**			**Filter name=FCBF**		
	J48	**Naïve**	**SMO**	**J48**	**Naïve**	**SMO**
Colon Tumor	87.09	85.48	87.09	90.32	85.48	87.09
Central Nervous System	78.33	76.66	85	78.33	76.66	85
Leukaemia	87.50	97.22	98.61	87.50	97.22	98.61
Breast Cancer	82.47	53.60	82.47	68.04	53.60	83.50
Ovarian Cancer	98.02	99.60	100	98.81	100	100
MLL	97.22	94.44	100	91.66	94.44	100
Lymphoma	96.96	100	100	96.96	100	100
Leukaemia-3C	84.72	97.22	97.22	86.11	94.44	95.83
Leukaemia-4C	87.50	93.05	94.44	83.33	91.66	91.66
SRBCT	85.54	100	100	85.54	100	98.79
Lung Cancer	92.11	96.05	96.05	91.62	96.05	95.56
Average	**88.86***	**90.30**	**94.62***	**87.11**	**89.95**	**94.18**

Table6. Classification results obtained by three learning algorithm against different subset that have obtained by GSNR, ReliefF and mRMR filters.

Dataset	**Filter name= GSNR**			**Filter name= ReliefF**			**Filter name=** mRMR		
	J48	**Naive**	**SMO**	**J48**	**Naive**	**SMO**	**J48**	**Naive**	**SMO**
Colon Tumor	80.64	83.87	87.09	67.74	72.58	85.48	79.03	87.09	87.09
Central Nervous System	65	75	78.33	73.33	76.66	80	73.33	75	81.66
Leukaemia	87.50	98.61	98.61	84.72	98.61	98.61	87.50	98.61	97.22
Breast Cancer	76.28	56.70	80.41	76.28	79.38	82.47	72.16	55.67	78.35
Ovarian Cancer	99.20	98.02	100	98.81	96.44	100	98.81	98.41	100
MLL	97.2	93.0	97.22	94.4	97.22	97.22	94.44	94.4	100

	2	5		4				4	
Lymphoma	96.96	100	100	92.42	100	100	87.87	100	100
Leukaemia-3C	86.11	98.61	98.61	87.50	98.61	97.22	84.72	98.61	95.83
Leukaemia-4C	90.27	94.44	94.44	90.27	93.05	91.66	90.27	94.44	91.66
SRBCT	87.95	100	100	87.95	100	98.79	87.95	98.79	100
Lung Cancer	91.62	83.74	88.67	85.22	90.64	94.08	92.61	95.07	96.05
Average	**87.15**	**89.27**	**93.03**	**85.33**	**91.19***	**93.23**	**86.24**	**90.55**	**93.44**

Table 7. Mean classification accuracy obtained by three learning algorithm against 11 gene datasets.

	CFS	FCBF	GSNR	ReliefF	mRMR
J48	88.86*	87.11	87.15	85.33	86.24
Naïve	90.30	89.95	89.27	91.19*	90.55
SMO	94.62*	94.18	93.03	93.23	93.44

As it can be seen in table 7, in general, the best result on j48 and SMO classifier has been achieved by CFS filter and on Naïve bayes, it has been achieved by ReliefF filter. Therefore, we use those subset filter that have obtained by these two filters.

Owing to the learning algorithm, Wrappers could achieve better feature subset in most cases, but we could not apply them on dataset at first, because the computational time and complexity would be unacceptable. Thus, after applying filters on datasets and having a new subset with small size of dimension, we apply wrappers on new feature subset in order to increase the accuracy. Table 8 shows the number of genes which are selected by wrappers for each individual microarray gene dataset.

Table 8. Number of selected genes for each wrappers.

Dataset	**Wrapper**			**Total features**
	J48	**Naive**	**SMO**	
Colon Tumor	3	6	7	11
Central Nervous System	2	12	10	17
Leukaemia	2	7	4	11
Breast Cancer	4	13	13	27
Ovarian Cancer	3	3	3	7
MLL	3	8	4	12
Lymphoma	3	4	3	7
Leukaemia-3C	4	7	5	12
Leukaemia-4C	6	8	6	15
SRBCT	5	6	9	14
Lung Cancer	8	11	13	25

Total features	**43**	**85**	**77**	**158***

As it can be seen in table 8, the number of genes which have been selected by wrappers is so lower than the number of initial gene. It should be mentioned that total features in table 8 refer to number of all features that have obtained for each wrapper (the last row) or each dataset (the last column) without considering common features. (For example in column1 of table8, total features are equal to 11 while we have 16 features (3+6+7) in overall. It shows that some features have been selected in more than 1 wrapper).

In order to evaluate the effectiveness of our method, we run our classifiers on new sets of genes which have achieved after three steps that described above. Then we combine the result according to the way that is described before. Table 9 summarizes the result of our approach on different feature sets in two columns (after and before applying wrappers). What is clear in table 9, the accuracy has increased when the wrappers applied on the feature sets which have obtained after applying filter.

Table 9. Accuracy before/after applying wrappers.

Dataset	**Accuracy before applying wrappers**	**Accuracy after applying wrappers**
Colon Tumor	88.70	90.32
Central Nervous System	83.33	93.33
Leukaemia	97.22	100
Breast Cancer	85.56	94.84
Ovarian Cancer	99.20	100
MLL	100	100
Lymphoma	100	100
Leukaemia-3C	97.22	98.61
Leukaemia-4C	93.05	100
SRBCT	100	98.79
Lung Cancer	96.05	96.55
Average	**94.57**	**97.49**

Finally, in order to compare our proposed algorithm, we have brought the result obtained of three researches that suggest an approach in order to overcome microarray data problem. All result shows in table 10.

Something that should be mentioned about table 10, is related to the triplet labeled ‘‘Win–Tie–Loss” in the last row of this table. The first value denotes the number of gene datasets on which our proposed method operates considerably better than the corresponding algorithm; the second value stands for the number of datasets on which the difference between the performance of our approach and that of the corresponding algorithm is not significant; and the third one indicates the number of datasets on which our proposed method performs significantly worse than the compared algorithm. As it can be seen, our approach is satisfactory against other approaches and it can be applied for high dimension data.

Table 10. Compare our approach with other references

Dataset	Our approach	Reference #[14]	Reference #[12]	Reference #[11]
Colon Tumor	90.32	85.66	-	**99.21**
Central Nervous System	93.33	72.21	**98.33**	90.19
Leukaemia	100	95.89	100	94.12
Breast Cancer	94.84	80.74	93.81	79.87
Ovarian Cancer	100	99.71	-	99.95
MLL	100	94.33	-	-
Lymphoma	100	97.68	100	96.13
Leukaemia-3C	98.61	96.64	-	-
Leukaemia-4C	100	91.93	-	-
SRBCT	98.79	**99.23**	-	-
Lung Cancer	96.55	**98.96**	89.58	**97.99**
Win/Tie/Loss	-	9/0/2	4/0/1	5/0/2

It should be mentioned that, all the algorithms were executed in MATLAB 7.12.0 (R2011a) and also we used a package namely "MATLAB Package for Gene Selection" [30].

5. CONCLUSIONS

In this work, we addressed an ensemble of filters and wrappers to cope with gene microarray classification problems. The idea is to utilize the efficiency of filters and the accuracy of wrappers. We also have used 3 type of classifier in order to recognize the power of each filters and set each classifier with one subset of feature. In classification stage, we applied all classifier instead of one classifier In order to take advantage of all classifier and suggested a good way in order to combine the results. Our approach was tested on 11 microarray dataset and the result show that our approach can be useful in microarray classification and we believe that our approach can be applicable in this such problem. Finally, we draw to this conclusion that even with low number of features you can achieve an acceptable accuracy as far as you have a strong feature selection step. In fact, this step plays an important role in classification problems.

REFERENCES

[1] Brown, P., Botstein, D., (1999) “Exploring the New World of the Genome with DNA Microarrays”, *Nature Genetics*, Vol. 21, pp. 33.

[2] Gordon, G., Jensen, R., Hsiao, L., Gullans, S., (2002) "Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests using Gene Expression Ratios in Lung Cancer and Mesothelioma", *Cancer Research*, Vol. 62, No. 17, pp. 4963-4971.

[3] Kohavi, R., John, G.H., (1997) "Wrappers for Feature Selection", *Artificial Intelligence*, Vol. 97, Nos. 1/2, pp. 273-324.

[4] Blanco, R., Larranaga, P., Inza, I., Sierra, B., (2004) "Gene Selection for Cancer Classification using Wrapper Approaches", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 18, pp. 1373-1390.

[5] Cho, S., Won, H., (2003) "Machine Learning in DNA Microarray Analysis for Cancer Classification", First Asia-Pacific Bioinformatics Conference on Bioinformatics, Vol. 19, pp. 189-198.

[6] Brown, M., et al., (2000) "Knowledge-Based Analysis of Microarray Gene Expression Data by using Support Vector Machines", *Proceedings of National Academy of Sciences*, Vol. 97, pp. 262-267.

[7] Furey, T., et al., (2000) "Support Vector Machine Classification and Validation of Cancer Tissue Samples using Microarray Expression Data", *Bioinformatics*, Vol. 16, No. 10, pp. 906-914.

[8] Friedman, N., et al., (2000) "Using Bayesian Networks to Analyze Expression data", *Fourth Annual International Conference on Computational Molecular Biology*, Vol. 7, No.3-4, pp. 127-135.

[9] Song, Q., Ni, J., Wang, G., (2013) "A Fast Clustering-Based Feature Subset Selection Algorithm for high dimensional data", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25, No. 1, pp. 1-14.

[10] Bolon-Canedo, V., Sanchez-Marono, N., Alonso-Betanzos, A., (2012) "An Ensemble of Filters and Classifiers for Microarray Data Classification", *Pattern Recognition*, Vol. 45, No. 1, pp. 531-539.

[11] Nagi, S., Bhattacharyya, D. Kr., (2013) "Classification of Microarray Cancer Data Using Ensemble Approach", *Network Modeling Analysis Health Informatics Bioinformatics*, Vol. 2, No. 3, pp.159–173.

[12] Liu, H., Liu, L., Zhang, H., (2010) "Ensemble Gene Selection for Cancer Classification", *Pattern Recognition*, Vol. 43, No. 8, pp. 2763–2772.

[13] Hala, M.A, Ghada, H.B, Yousef, A.A, (2015) "Genetic BeeColony(GBC) Algorithem: A new Gene Selection Method for Microarray Cancer Classification", *Computational Biology and Chemistry*, Vol. 56, pp. 49-60.

[14] Hall, M.A., (1999) *Correlation-Based Feature Selection for Machine Learning*, Ph.D. Thesis, University of Waikato, Hamilton, New Zealand.

[15] Zhu, Z., Ong, Y., Dash, M., (2007) "Markov Blanket-Embedded Genetic Algorithm for Gene Selection", Pattern Recognition, China. [Online] < csse.szu.edu.cn/staff/zhuzx/datasets.html>, [12 June 2014].

[16] Hall, M. A., (1999) *Correlation Based Feature Selection for Machine Learning*, Thesis Report, University of Waikato.

[17] Kononenko, I., (1994) "Estimating Attributes: Analysis and Extensions of RELIEF", *European Conference on Machine Learning*, Vol. 784, pp. 171–182.

[18] Kira, K., Rendell, L. M., (1992) "A Practical Approach to Feature Selection", *Proceedings of the Ninth International Workshop on Machine learning*, pp. 249–256.

[19] Ding, C., Peng, H., (2003) "Minimum Redundancy Feature Selection from Microarray Gene Expression Data", *2nd IEEE Computational Systems Bioinformatics*, pp. 523-528.

[20] Zheng, G., (2006) *Statistical Analysis of Biomedical Data with Emphasis on Data Integration*, Ph.D. Thesis, Florida International University.

[21] Yu, L., Liu, H., (2004) "Redundancy Based Feature Selection for Microarray Data", *KDD '04 Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, Research Track Poster, pp. 737-742.

[22] Al Snousy, M. B., El-Deeb, H. M., Badran, Kh., Al Khli, I. A., (2011) "Suite of Decision Tree-Based Classification Algorithms on Cancer Gene Expression Data", *Egyptian Informatics Journal*, Vol. 12, No. 2, pp.73–82.

[23] Hall, M.A., Smith, L.A., (1999) "Feature Selection for Machine Learning: Comparing a Correlation-Based Filter Approach to the Wrapper", *Proceedings of the Twelfth international Florida Artificial intelligence Research Society Conference*, pp. 235-239.

[24] Yu, L., Liu, H., (2003) "Feature Selection for High-Dimensional Data: a Fast Correlation-Based Filter Solution", *Proceedings of 20th International Conference on Machine Leaning*, Vol. 20, No.2, 856-863.

[25] Yu, L., Liu, H., (2004) "Efficient Feature Selection via Analysis of Relevance and Redundancy", *Journal of Machine Learning Research*, Vol. 10, No. 5, pp. 1205-1224.

[26] Zhao, Z., Liu, H., (2007) "Searching for Interacting Features", *Proceedings of the 20th International Joint Conference on AI*, pp. 1156-1161.

[27] Zhao, Z., Liu, H., (2009) "Searching for Interacting Features in Subset Selection", *Journal Intelligent Data Analysis*, Vol. 13, No. 2, pp. 207-228.

[28] Mateo, S., Kaufman, C. M., Quinlan J.R., (1993) *C4.5: Programs for Machine Learning*, Australia.

[29] Kuncheva, L., (2004) *Combining Pattern Classifiers(Methods and Algorithms)*, IEEE.

[30] Zhang, Y., (2009) "A Matlab Package for Gene Selection", Florida International University. [Online], <http://cs.fiu.edu/_yzhan004/genesel.html>, [12 June 2014].