An Embedded Two-Layer Feature Selection Approach for Microarray Data Analysis

Pengyi Yang and Zili Zhang

***Abstract*—Feature selection is an important technique in dealing with application problems with large number of variables and limited training samples, such as image processing, combinatorial chemistry, and microarray analysis. Commonly employed feature selection strategies can be divided into filter and wrapper. In this study, we propose an embedded two-layer feature selection approach to combining the advantages of filter and wrapper algorithms while avoiding their drawbacks. The hybrid algorithm, called GAEF (Genetic Algorithm with embedded filter), divides the feature selection process into two stages. In the first stage, Genetic Algorithm (GA) is employed to pre-select features while in the second stage a filter selector is used to further identify a small feature subset for accurate sample classification. Three benchmark microarray datasets are used to evaluate the proposed algorithm. The experimental results suggest that this embedded two-layer feature selection strategy is able to improve the stability of the selection results as well as the sample classification accuracy.**

***Index Terms*—Feature selection, Filter, Wrapper, Hybrid, Microarrays.**

I. Introduction

CURSE-OF-DIMENSIONALITY is a major problem associated with many classification and pattern recognition problems. When addressing the classification problems with a large number of features, the classifier created will often be very complex with poor generalization property. This is especially true in analyzing microarray datasets which inherently have several thousand of features (genes) with only a few dozen of samples [1]. One effective way to deal with such problems is to apply feature selection technologies [2]. The benefits of feature selection are as follows:

- Reducing the number of features to a sufficient minimum will cut the computational expenses.
- Feature selection can reduce the noise introduced in the classification process, which then will improve sample classification accuracy.
- From the biological perspective, minimizing feature size can help the researchers to concentrate on the selected genes for biological validation etc.
- The higher the ratio of the number of training sample to the number of features used by classifier, the better the generalization ability of the resulting classifier [3]. In other words, minimizing the size of the features can improve the generalization property of the resulting classification model.

Based on the selection manners, feature selection methods can be broadly divided into filter, wrapper and embedded approaches [4]. Among them, filter and wrapper approaches are the most popular ones in biological data analysis. Genetic Algorithm (GA), as an advanced type of wrapper selector, has been applied as the search scheme for microarray data analysis recently [5], [6], [7]. Unlike forward selection and backward elimination wrappers which select features linearly, GA selects features nonlinearly by creating feature combinations randomly. This character of GA accommodates the identification of the nonlinear relationship among features. Moreover, GA is efficient in exploring large feature space [8], [9], which makes it a promising solution for gene selection of microarray. However, as many wrapper selection strategies encountered, GA often suffers from overfitting [10] because an inductive algorithm is usually used as the sole criterion in feature subset evaluation. Another problem is that GA is unstable in feature selection because of its stochastic nature. Furthermore, GA is a near optimal search algorithm. This means when applying GA, we are facing the risk of trapping into local optimal solutions. This risk rises exponentially with the increase of the feature size.

Different from wrapper strategies, filter approaches do not optimize the classification accuracy of a given inductive algorithm directly. Instead, they try to select a feature set with a predefined evaluation criterion. Examples include t-test [11], χ^2-test [12], Information Gain [13] etc. Although filtering algorithms are superior in selecting of better generalization features which often extended well on unseen data, there are manifold disadvantages they suffered from. Firstly, filtering approaches totally ignore the effects of the selected feature subset on the performance of the inductive algorithm. However, the performance of the inductive algorithm may be crucial for accurate phenotype classification [14]. Secondly, filtering approaches are often deterministic and greedy based. This leads to only one feature profile being selected, which is often suboptimal, whereas a different feature profile may produce better classification results. Moreover, Jaeger et al. demonstrated that in microarray data analysis genes obtained by aggressive reduction with filter based methods are often highly correlated with each other, thus, redundant [15]. In classifier construction and sample classification, such a redundant feature set often increases the model complexity while decreases the generality [3].

In order to combine the strengths of filter and wrapper approaches while avoiding their drawbacks, we recently in-

Pengyi Yang is with School of Information Technologies (J12), The University of Sydney, NSW 2006, Australia.
E-mail: yangpy@it.usyd.edu.au

Zili Zhang is with Faculty of Computer and Information Science, Southwest University Chongqing 400715, China; School of Information Technology, Deakin University, Geelong, Victoria, Australia, 3217.
E-mail: zili.zhang@deakin.edu.au

troduced several hybrid feature selection strategies [16], [17]. In those studies, however, the filtering algorithm is used either as prior evaluator [16] or an intermediate scoring criteria [17]. In this study, we gives the filtering algorithm more control over the feature selection results and propose an embedded two-layer feature selection framework. The aim is to testify whether such formulation could improve sample classification accuracy and feature selection stability. This approach justifies its name because a filter algorithm is embedded in the GA algorithm. The embedded filter is used to evaluate and reduce the feature subsets randomly generated by GA and then feed the reduced subsets to the inductive algorithm for pattern recognition. Hence, the feature selection process is broken into two stages. We named it GAEF (Genetic Algorithm with embedded filter) for convenience. Different from many hybrid methods relying on manipulating learning datasets [18], this embedded two-layer feature selection model has following advantages:

- With the random selection of GA and the pattern recognition of the classifier, stochastic nature is integrated into the hybrid system as well as the performance information of the inductive algorithm.
- The unstable issue of GA is minimized because GA is designated to pre-select a very large feature subset while the final feature set is actually determined by the filter algorithm embedded in it.
- Since GA only “loosely” selects a large feature subset, the possibility of trapping into a suboptimal solution is minimized while generalization property is enhanced.
- The integration of the performance information of a given classifier in sample classification is used to minimize the correlation of the filter selected features implicitly, resulting in a redundancy reduced and information enriched feature subset.

Therefore, this GAEF algorithm is expected to possess more stable and generalization quality in feature selection, which contribute to a higher sample classification accuracy comparing with those obtained by applying its components alone. We apply the proposed method to three benchmark microarray datasets, including binary-class as well as multi-class classification problems. The empirical results obtained by using the proposed model are compared with those obtained by using GA wrapper and filter algorithms individually. Moreover, the classification results of a popular GA/KNN algorithm developed by Li et al. [5] for microarray data analysis are provided as the third yardstick. It’s worth noting that the proposed algorithm can also be applied to other feature selection domains such as image processing and combinatorial chemistry with minor modification.

The paper is organized as follows: In Section II, we present the overview of the proposed method. In Section III, the implementation and evaluation issues are detailed. Section IV provides the experimental results while Section V and Section VI discuss and conclude of the paper.

II. Embedded Two-Layer Feature Selection Approach

A. System Overview

From the data mining perspective, each sample in dataset is commonly described as a vector of the form $\mathbf{s}_i$=$[f_1, f_2, ..., f_n],\ (i = 1, ..., m)$, where m is the number of samples and n is the number of the features. The dataset is described as a $m \times n$ matrix D_{mn}=$\{(\mathbf{s}_1, y_1), (\mathbf{s}_2, y_2), (\mathbf{s}_m, y_m)\}$, where y_i is the class value of the ith sample. Feature selection is essentially to generate a reduced feature vector $\mathbf{s}'_i$=$[f_1, f_2, ..., f_d],\ (\mathbf{s}'_i \subset \mathbf{s}_i)$ which confines the dataset matrix into D_{md}=$\{(\mathbf{s}'_1, y_1), (\mathbf{s}'_2, y_2), (\mathbf{s}'_m, y_m)\}$ with the expectation to reduce the noisy and redundancy. The proposed GAEF approach utilizes a standard GA as the first layer of feature selection to generate and select large, pre-selected feature subsets $\mathbf{s}'_i$=$[f_1, f_2, ..., f_{d_1}],\ (\mathbf{s}'_i \subset \mathbf{s}_i)$. The embedded filter algorithm which serves as the second layer of feature selection is used to further determine a compact feature subset $\mathbf{s}''_i$=$[f_1, f_2, ..., f_{d_2}],\ (\mathbf{s}''_i \subset \mathbf{s}'_i)$ from each pre-selected feature subset of GA. Those further selected feature subsets are then fed into the classification algorithm for pattern recognition. For convenience, and without loss of generality, we simplify the notation of $\mathbf{s}'_i$ to s in the rest of the paper. Figure 1 illustrates the work flow of the GAEF model.

The algorithm performs following steps:

S1: Initially, GA randomly creates a set of chromosomes which representing various pre-selected feature subsets.

S2: Filter algorithm is invoked to select a further reduced feature subset from each pre-selected feature subset provided in GA chromosome.

S3: Feature sets selected by filter are then fed into classifier for sample classification and pattern recognition. After a classifier evaluates a given feature subset, it returns the classification strength of this feature subset to its corresponding pre-selected feature subset.

S4: After the whole population are evaluated, GA selects favorite chromosomes that can produce good feature subsets with a given filter in sample classification.

S5: The crossover and mutation operations are then conducted on the selected chromosomes with a predefined P_C (probability of crossover) and P_M (probability of mutation), respectively, and the next generation begins.

S6: Repeat steps 2-5 until terminating generation is reached and the final filter selected feature subsets are collected as the optimal feature profiles for sample classification and pattern recognition.

B. Subset Evaluation and Selection

In GA, the goodness of a candidate solution is evaluated by calculating a given fitness function using the bits configuration of this solution. In feature selection, such fitness function is often defined as the simple classification accuracy. However, the problem of using simple classification accuracy is that when the numbers of samples in different classes are imbalanced, the fitness score provided by such a measure could be misleading [19]. This can be shown with following examples. Suppose a binary-class dataset contains 5 samples from class A and 45

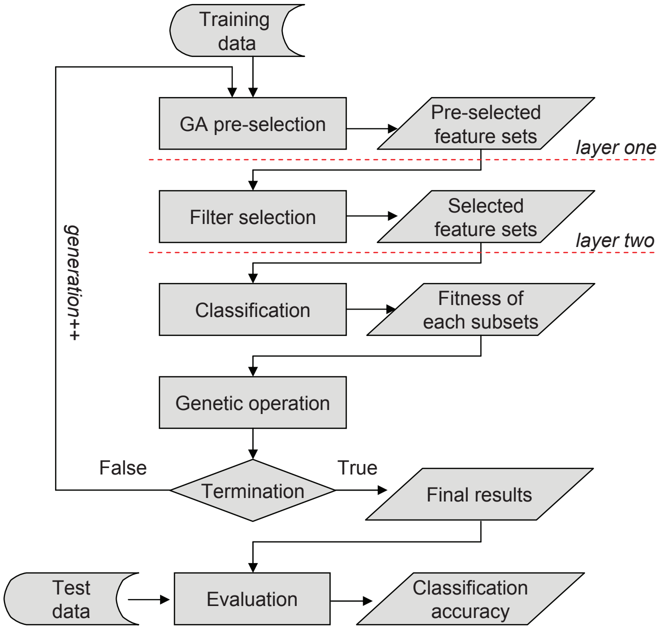

Fig. 1. GAEF work flow. GA is used to produce large, pre-selected candidate feature sets and a filter algorithm is invoked to select a compact feature set from those pre-selected sets for sample classification.

samples from class B. If a classifier misclassifies all samples in class A but correctly classifies other 45 samples in class B, the fitness score produced by simple classification accuracy measure is $45/50 \times 100 = 90\%$. However, no differential pattern is actually identified by the classifier, and the resulting feature subset is in fact useless for sample separation of unseen data. The problem worsen if the dataset at hand is multi-class. To overcome such problems, we utilized a balanced classification accuracy for feature subset evaluation and fitness calculation. Fitness function derived from such a balanced classification accuracy is defined as:

$$fitness(s) = \frac{\sum_{i=1}^{c} Se_i}{c} \qquad (1)$$

where c denotes the number of classes in the dataset, and s denotes the subset under evaluation. Se_i denotes the classification sensitivity of the samples in class i, which is calculated as follows:

$$Se_i = \frac{N_i^{TP}}{N_i} \times 100, \qquad (2)$$

where N_i^{TP} denotes the number of true positive classification of samples in class i, and N_i denotes the total number of samples in class i. For previous example, the fitness score given by this balanced accuracy measure is $(0/5+45/45)/2 = 50\%$. This result is significantly lower than that of simple classification accuracy measure which helps to correct the fitness score.

Followed by subset evaluation, tournament selection strategy is used for the selection of favorite chromosomes. In tournament selection, larger tournament size gives faster convergence speed of GA, and we found three member tournament selection is a good trade-off. Formally, the winner is determined as follows:

$$Winner = \arg\max_{s \in S} fitness_i(R(s)) \quad (i = 1, 2, 3) \qquad (3)$$

where $R(.)$ is the random function which randomly selects feature subset from the population S of GA, while $fitness(.)$ determines the fitness of the randomly selected feature subsets.

C. Filters

χ^2-test and Information Gain are popular filtering algorithms and are commonly used in gene selection of microarrays [12], [13]. We used this two types of filtering algorithms for forming the proposed hybrid algorithm, respectively. When used for feature selection purpose, χ^2-test can be considered as to evaluating occurrence of certain value of a feature and occurrence of the class. The feature is then ranked with respect to the following quantity:

$$\chi^2(f) = \sum_{v \in V} \sum_{i=1}^{m} \frac{(N(f = v, c_i) - E(f = v, c_i))^2}{E(f = v, c_i)} \qquad (4)$$

where c_i, $(i = 1, ..., m)$ denotes the possible classes of the dataset, while f is the feature that has a set of possible values denoted as V. $N(f = v, c_i)$ and $E(f = v, c_i)$ are the observed and the expected co-occurrence of $f = v$ with the class c_i, respectively.

Information Gain is another type of statistic measure for feature selection. It measures the number of bits of information provided in class prediction by knowing the value of feature. Again, let c_i belong to a set of discrete classes (1, ...,m).

V be the set of possible values for candidate feature f. The information gain of a feature f is then defined as follows:

$$\begin{aligned} Gain(f) = &-\sum_{i=1}^{m} P(c_i)\log P(c_i) \\ &+\sum_{v\in V}\sum_{i=1}^{m} P(f=v)P(c_i|f=v)\log P(c_i|f=v) \end{aligned} \quad (5)$$

D. Classification

kNN is a relatively computational efficient classifier which has been applied by several studies in evaluating gene selection [5], [6]. It calculates the similarity, called the distance, of a given instance with others and assign the given sample into the class to which the k most similar samples belong. Such a similarity can be defined as Euclidean distance, Manhattan distance or Pearson's correlation etc. We utilized k-Nearest Neighbor (kNN) classifier for sample classification and evaluation of the "merits" of feature subsets. In our GAEF algorithm, Euclidean distance is used for sample similarity comparison. Formally:

$$ED(\mathbf{x_1}, \mathbf{x_2}) = \sqrt{\sum_{i=i}^{d}(x_1(f_i) - x_2(f_i))^2}, \qquad (f_i \in s) \quad (6)$$

where $\mathbf{x_1}$ and $\mathbf{x_2}$ are two samples described by the subset s which is a feature vector $[f_1, f_2, ...f_d]$.

III. Experimental Settings

A. Datasets

Microarray technologies make parallel evaluation of several thousand of genes possible. On the contrary, the samples collected for such evaluation are often with limited size–a few dozen. Therefore, most microarray datasets are with large number of gene features and limited number of samples, which make them ideal for the evaluation of the proposed algorithm. In the initial experiment, we evaluated the proposed method with three benchmark microarray datasets. The first two, namely "Colon" and "Breast", are binary-class datasets, which are generated from microarray studies of colon cancer [20] and breast cancer [21], respectively. The third microarray dataset called "MLL" is a multi-class dataset generated from a leukemia study [22]. Table I summarizes each dataset.

TABLE I
Microarray datasets used in evaluation

Name	Colon	Breast	MLL
No. of Gene	2000	24481	15154
No. of Sample	62	97	72
No. of Class	2	2	3
C1:	Normal (22)	Relapse (46)	ALL (24)
C2:	Cancer (40)	Non-relapse (51)	MLL (20)
C3:			AML (28)

Expression values of each gene in each dataset are normalized into [0,1] with the mean of 0 and the variance of 1 before feeding for pattern recognition. As to the Breast and the Prostate datasets, for the purpose of computational efficiency, we conducted a Symmetrical Uncertainty analysis [23] to reduce the feature dimension from 24481 to 2000 and from 15154 to 2000, respectively.

B. GAEF Implementation

A standard GA is used in GAEF implementation as the first layer of feature selection. The population size of GA is set to 100. We adopt single point crossover and mutation, with the probability of 0.6 and 0.02, respectively as they produced good classification results. Three members tournament selection strategy is utilized for favorite chromosome selection. We implemented three termination conditions. The first condition is that the algorithm reaches the 50th generation. The second one requires that the chromosomes in a GA generation converge to 90%. The last condition is that no fitness improvement is generated in the last 5 sequential GA generations.

As to the GA pre-selection size, after some preliminary test we decide to fix it to 400 genes as it produces good experimental results. In regard to the second gene selection layer, we examined χ^2-test and Information Gain algorithms. By exploring combining different filters, we are able to evaluate the generality of the proposed embedded two-layer feature selection model. Based on the previous study [24], in most cases only a few dozen (or a few) genes are needed for sample classification. Therefore, we vary the embedded filter selection of the gene sizes from 5 to 25 with a step of 5. Lastly, each gene subset is evaluated by kNN classifier. Previous studies, demonstrated that small values of k such as odd number of 3 and 5 often produce good classification results [5]. In our experiments, $k = 3$ is arbitrarily chosen.

Table II summarizes the parameter setting of the GAEF model.

TABLE II
GAEF parameter settings

Parameter	**Value**
Genetic Algorithm	Single Objective
Population Size	100
Chromosome Size	400
Selector	Tournament Selection
Crossover	Single Point (0.6)
Mutation	Single Point (0.02)
Termination Condition	Multiple Condition
Candidate Filter	χ^2-test; InfoGain
Filtering Size	5 to 25 (step of 5)
Inductive Algorithm	kNN

C. Correlation Evaluation

As pointed out by Jaeger et al. [15], in microarray study genes obtained by aggressive reduction with filter based methods are often highly correlated which inevitably introduce noisy and redundancy. Therefore, several studies attempted to minimize the correlation of selected genes to the minimum [25], [26]. However, those measures try to get rid of correlation in the selected gene subset all together, while such correlation information may not be totally uninformative. For example, in study [27], Xu and Zhang suggested that such correlation itself may be used as predictor of sample class.

In our algorithm, the correlation of selected genes is minimized in a more moderate manner. That is, through the use of an inductive algorithm the correlation of the selected genes is minimized implicitly. Our objective is to minimize the redundancy while keeping the usefulness. After all, the reason of minimizing gene correlation is to obtain higher classification accuracy. In our experiment, we compare the correlation of the most frequently selected genes using the proposed method to those obtained by using filter algorithms directly. The calculation of the average correlation is as follows:

$$P(x_i, x_j) = \frac{\sum x_i x_j - \frac{(\sum x_i)(\sum x_j)}{m}}{\sqrt{(\sum x_i^2 - \frac{(\sum x_i)^2}{m})(\sum x_j^2 - \frac{(\sum x_j)^2}{m})}} \qquad (7)$$

$$Average\ Correlation = \frac{2 \times \sum_{i=1}^{n} \sum_{j=i+1}^{n} \sqrt{P(x_i, x_j)^2}}{n(n-1)} \qquad (8)$$

where x_i and x_j denote the expression level of two different genes in the selection result. $P(.)$ is the function of Pearson Product-Moment correlation coefficient. m denotes the total samples, while n denotes the total number of genes considered.

D. Cross Validation and Stability

Cross validation is one of the most popular evaluation strategies. When employing cross validation, the dataset is commonly divided into several folds. Taking n-fold cross validation as an example, while $n - 1$ folds are used to train the classifier the remaining fold is used to evaluate the classification power of the classifier on unseen data. After each fold is used to evaluate the classification accuracy in an orderly fashion, the classification accuracy of the classifier is then calculated by averaging the classification accuracy of each fold. Cross validation is a robust evaluation method. It is particularly useful when the sample size of the dataset is small because the dataset is efficiently reused for measuring the error rate, resulting a more objective evaluation results [28]. In this work, 5-fold stratified cross validation is utilized.

With the consideration of the stochastic nature of GA, each GA based method is conducted with 5 independent runs, producing 5 independent cross validation results. The final results are given in the form of "mean ± standard deviation" ($\mu \pm \sigma$). The stability of each GA based method can then be assessed by comparing the value of the standard deviation σ.

IV. Results

A. Sample Classification Accuracy

For comparison purpose, we experimented using GA wrapper and filters (χ^2-test and Information Gain) separately for feature selection. The classification results of each individual selection method is compared with those obtained with GAEF.

Tables III–V give classification accuracy details of each method, using Colon, Breast and MLL microarray datasets, respectively [20], [21], [22]. Specifically, the second and the third columns of each table detail the sample classification using χ^2-test and Information Gain selected gene sets (from size of 5 to 25 with a step of 5) with kNN classifier. The fourth column shows the classification of GA wrapper selected gene sets with kNN classifier. And the last two columns provide the classification results obtained by using GAEF selected gene sets with embedded filters of χ^2-test and Information Gain, respectively. Each GA based selection method is averaged with 5 independent runs.

As can be readily observed, in most cases GAEF identified gene subsets produced better sample classification results. With Colon dataset, using GAEF selected gene sets we obtained the average sample classification accuracy of 83.36 and 82.46 using embedded filters of χ^2-test and Information Gain, respectively. Compared with using these two filter algorithms directly, which produced the average classification accuracy of 73.67 and 75.98, the improvement is significant. Similar results can be observed in the analysis results of both Breast dataset and MLL dataset. The classification accuracy of GAEF identified gene subsets for Breast dataset are 68.03 and 67.01, while for MLL dataset the figures are 86.28 and 87.89, using χ^2-test and Information Gain, respectively. In comparison, the classification accuracy produced with the two filter algorithms directly are 63.69 and 63.54 for Breast dataset, and 82.69 and 82.15 for MLL dataset. Although not so phenomenal compared with that of Colon dataset, the improvement is still obvious. Essentially, χ^2-test and Information Gain produced similar classification results regardless been used solely or embedded in GA. By applying GA wrapper directly for gene subsets selection, the average classification accuracy are 71.79 for Colon data, 64.46 for Breast data and 82.06 for MLL data. The results are similar to those achieved by applying filter based gene selection and sample classification.

With regard to the stability of the classification results, when applying GA wrapper directly, the variance σ is usually quite large, which is consistent with our assumption that GA is unstable and prone to local optimal with high feature-to-sample ratio data. This phenomenon is evident from the analysis results of all of the three microarray datasets. For Colon, Breast and MLL datasets, the average variance of the classification results are 5.29, 5.02 and 3.52, respectively (column 4 of Tables III–V).

In contrast, results yielded by using GAEF model are with smaller variance (column 5 and 6 of Tables III–V). With Colon dataset, the average variance of the classification result is 2.47 for the χ^2-test embedded model and 2.45 for the Information Gain embedded model. With Breast dataset, the average variance of the classification result is 2.77 for the χ^2-test embedded model and 2.82 for the Information Gain embedded model. As to the MLL dataset, the figures are 2.05 and 2.50 for the χ^2-test embedded model and the Information Gain embedded model, respectively. These results suggest that by adding an embedded filter, we are able to improve the stability of GA based feature selection algorithms.

B. Comparison of GA/KNN

Table VI provides the 5-fold stratified cross validation results utilizing kNN with the gene sets identified by GA/KNN algorithm [5], using identical divisions of training and test sets as that of GAEF. When applying GA/KNN, the chromosome length of 10 is used, and the number of near-optimal combinations selected is 1000. Majority voting and the $k = 3$ of

TABLE III
5-FOLD STRATIFIED CROSS VALIDATION ACCURACY OF COLON DATASET

				GAEF	
Feature size	χ^2+kNN	Info+kNN	GA+kNN	GA+χ^2+kNN	GA+Info+kNN
5-gene	71.22	72.11	70.55 ± 5.79	81.22 ± 2.71	80.53 ± 2.17
10-gene	72.55	76.11	73.37 ± 5.40	84.62 ± 2.61	81.82 ± 2.36
15-gene	73.26	73.89	71.07 ± 5.96	83.02 ± 1.55	83.55 ± 3.44
20-gene	76.22	78.89	69.38 ± 4.47	83.58 ± 2.61	83.78 ± 0.67
25-gene	75.11	78.89	74.60 ± 4.82	84.34 ± 2.89	82.60 ± 3.61

TABLE IV
5-FOLD STRATIFIED CROSS VALIDATION ACCURACY OF BREAST DATASET

				GAEF	
Feature size	χ^2+kNN	Info+kNN	GA+kNN	GA+χ^2+kNN	GA+Info+kNN
5-gene	57.14	55.34	64.24 ± 4.70	66.07 ± 4.49	62.51 ± 2.66
10-gene	66.11	62.54	62.58 ± 4.39	70.17 ± 3.08	66.75 ± 3.89
15-gene	68.29	66.30	64.44 ± 5.67	66.56 ± 1.92	68.99 ± 3.69
20-gene	61.99	64.28	65.61 ± 5.86	67.51 ± 1.80	69.61 ± 1.15
25-gene	64.96	69.24	65.43 ± 4.46	69.85 ± 2.58	67.20 ± 2.72

TABLE V
5-FOLD STRATIFIED CROSS VALIDATION ACCURACY OF MLL DATASET

				GAEF	
Feature size	χ^2+kNN	Info+kNN	GA+kNN	GA+χ^2+kNN	GA+Info+kNN
5-gene	80.00	79.33	74.44 ± 4.53	84.47 ± 1.58	88.31 ± 2.93
10-gene	84.00	81.11	83.74 ± 4.27	86.84 ± 1.59	86.29 ± 1.07
15-gene	83.11	81.11	85.64 ± 2.16	85.49 ± 2.39	87.64 ± 3.45
20-gene	82.11	85.78	80.87 ± 5.39	87.69 ± 2.56	87.00 ± 1.32
25-gene	84.22	83.44	85.60 ± 1.24	86.89 ± 2.13	90.20 ± 3.74

the k-nearest neighbor are adopted. It should be noted that the cut off of the selection threshold for the chromosomes of GA/KNN depends on the characteristics of the datasets. Different thresholds are used according to its classification power on different datasets. Specifically, the threshold for the Colon dataset is that 4 samples are incorrectly classified at most. For Breast dataset and MLL dataset the thresholds are 5 and 2 samples are incorrectly classified at most.

Comparing the results produced by our GAEF method with those obtained from GA/KNN algorithm, we can conclude that GAEF method is comparable or even superior in several cases to GA/KNN algorithm in terms of gene selection for microarray data classification.

TABLE VI
CLASSIFICATION ACCURACY OF GA/KNN ALGORITHM

GA/KNN			
Feature size	Colon	Breast	MLL
5-gene	74.78	66.63	86.22
10-gene	76.55	68.63	87.89
15-gene	83.11	69.81	85.45
20-gene	83.11	69.40	88.11
25-gene	83.11	69.36	87.00

C. Correlation of Frequently Identified Genes

Tables 5-7 give the top-5 most frequently selected genes of Colon dataset, Breast dataset and MLL dataset, respectively. Specifically, Hsa.37937 and Hsa.692 in Colon dataset, Contig7258_RC in Breast dataset, and 40763_at, 32847_at and 35614_at in MLL dataset are the most frequently selected genes using different methods. Each table is subdivided into four sub-tables corresponding to the gene selection methods of using χ^2-test and Information Gain directly, and using they as GA embeds. Selected genes in each sub-table are pairwised with each other for Pearson Product-Moment correlation coefficient calculation. It is evident that the average Pearson correlation coefficients of GAEF selected genes are generally lower than those identified directly by filter algorithms. Nevertheless, GAEF algorithm did not attempt to reduce the correlation between each pair of genes to the minimum. This is because as demonstrated in empirical study [27] correlation among genes does not necessarily be totally useless. On the contrary, it may facilitate the sample classification in some degree.

V. DISCUSSION

One major problem of applying GA based wrapper for feature selection of high dimensional dataset is that the algorithm is prone to overfitting and often quickly converge to a local optimal solution. Therefore, the selected feature subsets often perform poor on unseen data classification. This phenomenon is evident in our experimental results that using GA with kNN classifier for gene selection and data classification of microarrays. By embedding an filtering algorithm into the GA wrapper, we are able to minimize the overfitting of the resulting hybrid algorithm in feature selection and sample classification processes. The explanation of this improvement is straightforward. By adding a filter algorithm, candidate

TABLE VII
TOP-5 MOST FREQUENTLY SELECTED GENES OF COLON DATASET AND THEIR PAIRWISE CORRELATIONS

χ^2+kNN

Gene id	Hsa.627	Hsa.8147	Hsa.37937	Hsa.692(f765)	Hsa.1832
Hsa.627	-				
Hsa.8147	-0.277	-			
Hsa.37937	-0.315	0.815	-		
Hsa.692(f765)	-0.298	0.794	0.761	-	
Hsa.1832	-0.283	0.815	0.886	0.725	-
Average Pearson correlation coefficient: 0.597					

Info+kNN

Gene id	Hsa.627	Hsa.8147	Hsa.37937	Hsa.692(f765)	Hsa.692(f267)
Hsa.627	-				
Hsa.8147	-0.277	-			
Hsa.37937	-0.315	0.815	-		
Hsa.692(f765)	-0.298	0.794	0.761	-	
Hsa.692(f267)	-0.285	0.886	0.739	0.851	-
Average Pearson correlation coefficient: 0.602					

GA+χ^2+kNN

Gene id	Hsa.692(f267)	Hsa.37937	Hsa.601	Hsa.692(f765)	Hsa.3306
Hsa.692(f267)	-				
Hsa.37937	0.739	-			
Hsa.601	-0.243	-0.237	-		
Hsa.692(f765)	0.851	0.761	-0.279	-	
Hsa.3306	-0.223	-0.147	0.665	-0.189	-
Average Pearson correlation coefficient: 0.433					

GA+Info+kNN

Gene id	Hsa.5971	Hsa.41323	Hsa.692(f245)	Hsa.2451	Hsa.2291
Hsa.5971	-				
Hsa.41323	0.705	-			
Hsa.692(f245)	-0.192	-0.120	-		
Hsa.2451	0.567	0.582	-0.155	-	
Hsa.2291	-0.178	-0.012	0.571	0.145	-
Average Pearson correlation coefficient: 0.323					

features are no longer evaluated by the sole criterion of the classification accuracy of a given inductive algorithm, but regulated by the filter algorithm implicitly. Hence, the hybrid algorithm itself does not seek for high classification accuracy of training dataset blindly and greedily but take into consideration of other characteristics of the data as well. In this way, different selection criteria are balanced, and the "importance" of a given feature to the dataset is evaluated from multiple aspects.

VI. CONCLUSIONS

Filter and wrapper algorithms are commonly treated as competitors in feature selection of datasets with high dimension. Several studies have been conducted to compare the strengths and the weaknesses of each method in microarray data analysis context [12], [29], [30], but few of them attempted to integrate individual methods. In this study, instead of treating each method as competitor, we take the effort to integrate them as components of a higher system. The proposed hybrid model called GAEF utilizes GA to pre-select large feature subsets and invokes a filter selector to further identify highly differential feature subsets for accurate sample classification. This model is tested on both binary-class dataset and multi-class dataset. The experimental results suggest that such an embedded two-stage feature selection model be able to improve sample classification accuracy as well as the stability of the selection results.

ACKNOWLEDGMENT

We would like to thank Chuan Cao from Thinkit Speech Lab., Institute of Acoustics, Chinese Academy of Sciences for valuable discussion and suggestion.

REFERENCES

[1] R.L. Somorjai, B. Dolenko and R. Baumgartner, "Class prediction and discovery using gene microarray and protenomics mass spectroscopy data: curses, caveats, cautions," *Bioinformatics*, vol. 19, pp. 1484-1491, 2003.

[2] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine learning Research*, vol. 3, pp. 1157-1182, 2003.

[3] S. Theodoridis and K. Koutroumbas, *Pattern Recognition (Third Edition)*. Elsevier Press, 2006.

[4] A.L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, vol. 97, pp. 245-271, 1997.

[5] L. Li, C.R. Weinberg, T.A. Darden and L.G. Pedersen, "Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method," *Bioinformatics*, vol. 17, pp. 1131-1142, 2001.

[6] T. Jirapech-Umpai and S. Aitken, "Feature selection and classification for microarray data analysis: evolutionary methods for identifying predictive genes," *BMC Bioinformatics*, vol. 6, pp. 148, 2005.

[7] P. Yang and Z. Zhang, "Hybrid Methods to Select Informative Gene Sets in Microarray Data Classification," In: *Proceedings of the 20th Australian Joint Conference on Artificial Intelligence, LNAI 4830*, pp. 811-815, 2007.

[8] M. Kudo and J. Sklansky, "Comparison of algorithms that select features for pattern classifiers," *Pattern Recognition*, vol. 33, pp. 25-41, 2000.

[9] L. Kuncheva and L. Jain, "Designing classifier fusion system by genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4, pp. 327-336, 2000.

TABLE VIII
TOP-5 MOST FREQUENTLY SELECTED GENES OF BREAST DATASET AND THEIR PAIRWISE CORRELATIONS

χ^2+kNN					
Gene id	Contig31033_RC	Contig24311_RC	Contig15031_RC	Contig7258_RC	Contig30098_RC
Contig31033_RC	-				
Contig24311_RC	-0.069	-			
Contig15031_RC	0.679	0.175	-		
Contig7258_RC	-0.071	0.999	0.175	-	
Contig30098_RC	-0.305	0.277	-0.233	0.476	-
Average Pearson correlation coefficient: 0.346					

Info+kNN					
Gene id	Contig31033_RC	Contig7258_RC	Contig24311_RC	Contig15031_RC	NM_003344
Contig31033_RC	-				
Contig7258_RC	-0.071	-			
Contig24311_RC	-0.069	0.999	-		
Contig15031_RC	0.679	0.175	0.175	-	
NM_003344	0.522	0.154	0.156	0.531	-
Average Pearson correlation coefficient: 0.353					

GA+χ^2+kNN					
Gene id	AL080059	NM_020974	Contig7258_RC	AF073519	NM_014554
AL080059	-				
NM_020974	-0.462	-			
Contig7258_RC	0.458	-0.602	-		
AF073519	0.247	-0.414	0.391	-	
NM_014554	-0.130	-0.008	-0.204	0.007	-
Average Pearson correlation coefficient: 0.292					

GA+Info+kNN					
Gene id	NM_006115	NM_005744	NM_003258	Contig52554_RC	NM_016185
NM_006115	-				
NM_005744	0.363	-			
NM_003258	0.518	0.307	-		
Contig52554_RC	0.056	-0.248	-0.206	-	
NM_016185	0.360	0.373	0.692	-0.174	-
Average Pearson correlation coefficient: 0.329					

[10] Y. Saeys, I. Inza and P. Larranage, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, pp. 2507-2517, 2007.

[11] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield and E. Lander, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science*, vol. 286, pp. 531-537, 1999.

[12] H. Liu, J. Li and L. Wang, "A Comparative Study on Feature Selection and Classification Methods Using Gene Expression Profiles and Proteomic Patterns," *Genome Informatics*, vol. 13, pp. 51-60, 2002.

[13] Y. Su, T. Murali, V. Pavlovic, M. Schaffer and S. Kasif, "RankGene: Identification of Diagnostic Genes Based on Expression Data," *Bioinformatics*, vol. 19, pp. 1578-1579, 2003.

[14] R. Kohavi and G. John, "Wrapper for feature subset selection", *Artificial Intelligence*, vol. 97, pp. 273-324, 1997.

[15] J. Jaeger, R. Sengupta, W. Ruzzo, "Improved Gene Selection for Clssification of Microarrays," *Pacific Symposium on Biocomputing*, vol. 8, pp. 53-64, 2003.

[16] P. Yang, B. Zhou, Z. Zhang and A. Zomaya, " A multi-filter enhanced genetic ensemble system for gene selection in microarray data," to appear in APBC 2010.

[17] Z. Zhang, P. Yang, X. Wu and C. Zhang, "An agent-based hybrid system for microarray data analysis," *IEEE Intelligent Systems*, vol. 24, no. 5, pp. 53-63, 2009.

[18] S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection," In: *Proceedings of 18th International Conference on Machine Learning*, pp. 74-81, 2001.

[19] T. Khoshgoftaar, C. Seiffert and J. Hulse, "Hybrid Sampling for Imbalanced Data," In: *Proceedings of the IEEE International Conference on Information Reuse and Integration*, pp. 202-207, 2008.

[20] U. Alon, N. Barkai, D. Notterman, K. Gish, S. Ybarra, D. Mack and A. Levine, "Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays," *PNAS*, vol. 96, pp. 6745-6750, 1999.

[21] L. van't Veer, H. Dai, M. van de Vijver, Y. He, A. Hart, M. Mao, H. Peterse, K. van der Kooy, M. Marton, A. Witteveen, G. Schreiber, R. Kerkhoven, C. Roberts, P. Linsley, R. Bernards and S. Friend, "Gene expression profiling predicts clinical outcome of breast cancer," *Nature*, vol. 415, pp. 530-536, 2002.

[22] S. Armstrong, J. Staunton, L. Silverman, R. Pieters, M. den Boer, M. Minden, S. Sallan, E. Lander, T. Golub and S. Korsmeyer, "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature Genetics*, vol. 30, pp. 41-47, 2001.

[23] I. Witten and M. Frank, *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*. Elsevier, 2005.

[24] J. Hua, Z. Xiong, J. Lowey, E. Suh and E. Dougherty, "Optimal number of features as a function of sample size for variuos classification rules," *Bioinformatics*, vol. 21, pp. 1509-1515, 2005.

[25] Z. Cai, R. Goebel, M. Salavatipour and G. Lin, "Selecting dissimilar genes for multi-class classification, an application in cancer subtyping," *BMC Bioinformatics*, vol. 8, pp. 206, 2007.

[26] B. Hanczar, M. Courtine, A. Benis, C. Hennegar, K. Clement and J. Zucker, "Improving classification of microarray data using prototype-based feature selection," *SIGKDD Explorations*, vol. 5, pp. 23-30, 2003.

[27] X. Xu and A. Zhang, "Virtual gene: Using correlations between genes to select informative genes on microarray datasets," *Transaction on Computational System Biology II, LNBI 3680*, pp. 138-152, 2005.

[28] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, vol. 2, pp. 1137-1143, 1995.

[29] I. Inza, P. Larranage, R. Blanco and A. Cerrolaza, "Filter versus wrapper gene selection approaches in DNA microarray domains," *Artificial Intelligence in Medicine*, vol. 31, pp. 91-103, 2004.

[30] J. Lee, J. Lee, M. Park and S. Song, "An extensive comparison of recent classification tools applied to microarray data," *Computational Statistics & Data Analysis*, vol. 48, pp. 869-885, 2005.

TABLE IX
TOP-5 MOST FREQUENTLY SELECTED GENES OF MLL DATASET AND THEIR PAIRWISE CORRELATIONS

χ^2+kNN					
Gene id	36239_at	35164_at	32847_at	40763_at	39318_at
36239_at	-				
35164_at	0.580	-			
32847_at	0.712	0.745	-		
40763_at	-0.159	-0.152	-0.193	-	
39318_at	0.626	0.578	0.629	-0.142	-
Average Pearson correlation coefficient: 0.452					

Info+kNN					
Gene id	36239_at	35164_at	32847_at	40763_at	37539_at
36239_at	-				
35164_at	0.580	-			
32847_at	0.712	0.745	-		
40763_at	-0.159	-0.152	-0.193	-	
37539_at	0.688	0.625	0.669	-0.152	-
Average Pearson correlation coefficient: 0.468					

GA+χ^2+kNN					
Gene id	31886_at	41747_s_at	35164_at	266_s_at	40763_at
31886_at	-				
41747_s_at	0.304	-			
35164_at	0.356	0.457	-		
266_s_at	0.602	0.528	0.650	-	
40763_at	-0.109	0.136	-0.152	-0.163	-
Average Pearson correlation coefficient: 0.346					

GA+Info+kNN					
Gene id	36122_at	32847_at	35260_at	35614_at	1914_at
36122_at	-				
32847_at	0.413	-			
35260_at	0.494	0.732	-		
35614_at	0.334	0.661	0.738	-	
1914_at	-0.169	-0.279	-0.170	-0.213	-
Average Pearson correlation coefficient: 0.420					