Building a Classifier for Integrated Microarray Datasets through Two-Stage Approach

Youngmi Yoon, Jongchan Lee, and Sanghyun Park

Abstract— **Since microarray data acquire tens of thousands of gene expression values simultaneously, they could be very useful in identifying the phenotypes of diseases. However, the results of analyzing several microarray datasets which were independently carried out with the same biological objectives, could turn out to be different. One of the main reasons is attributable to the limited number of samples involved in one microarray experiment. In order to increase the classification accuracy, it is desirable to augment the sample size by integrating and maximizing the use of independently-conducted microarray datasets. In this paper, we propose a two-stage approach which firstly integrates individual microarray datasets to overcome the problem caused by limited number of samples, and identifies informative genes, secondly builds a classifier using only the informative genes. The classifier from large samples by integrating independent microarray datasets achieves high accuracy, sensitivity, and specificity on independent test sample dataset.**

Index Terms—**Bioinformatics, Microarray data analysis, Microarray data Integration, Microarray classification, Informative gene selection**

I. Introduction

RECENTLY, researchers have examined the gene expression pattern which is specific to tumor-cell and made use of molecular characteristics of tumor tissue for diagnosis purpose. Since microarray technology is capable of screening thousands of genes simultaneously, it is expected that microarray data will bring a drastic advancements in the field of tumor diagnosis.

As shown in Fig. 1, microarray data are organized as matrices such that each column represents a sample, each row represents a gene, and each cell represents the expression value of a particular gene in a particular sample. Since simultaneous measurements of expression levels for several tens of thousands of probes are now feasible, a statistical methodology is required for analysis and interpretation of a large volume of data.

This work was supported by grant No.(R01-2006-000-11106-0) from the Basic Research Program Korea Science and Engineering Foundation of Ministry of Science & Technology.

Youngmi Yoon is with the Department of Computer Science, Yonsei University, Korea (e-mail: amyyoon@cs.yonsei.ac.kr).

Jongchan Lee is with the Department of Computer Science, Yonsei University, Korea (e-mail: jclee@cs.yonsei.ac.kr).

Sanghyun Park is with the Department of Computer Science, Yonsei University, Korea (corresponding author to provide phone: 82-2-2123-5714; fax: 82-2-365-2579; e-mail: sanghyun@cs.yonsei.ac.kr).

		C_1			C_2	
	S_1	S_2	S_3	S_4	S_5	S_6
G_1	3	5	7	9	11	13
G_2	15	32	23	12	2	3
G_3	…	…	…	…	…	…
G_4						
G_5						
G_6						

Fig. 1. A microarray data
(S_i is a sample, G_i is a gene, and C_1, C_2 is a class label each)

When a statistical method is employed, increasing sample size is quite desirable for more reliable classification results. Especially in the tumor related research, the analysis with a large number of samples is quite essential in order to deduce a meaningful conclusion from data. Recently, Rhodes[13] has proposed a meta-analysis of multiple datasets that address similar hypotheses in order to validate and statistically assess all of the positive results simultaneously.

Considering only the microarray data with the same experimental objectives, differences in microarray platform, set of genes, technology and protocols used in different labs, still lead to difficulties in integrating microarray data across experiments. How to combine the data(gene expression levels) in different microarrays is a challenging problem, because these gene expression levels are not necessarily directly comparable. In this regard, we propose a method to integrate independent microarray datasets, and build a classifier through two stages.

In the first stage, we apply the integration algorithms combined with filtering methods to select a set of informative genes. Our integration algorithms do not require massive computation for normalization. Our informative gene filtering algorithm is a rank based approach within a sample. In the second stage, we build a classifier using only the pre-selected informative genes, and the biological interpretation of the classifier is relatively simple. Our classifier consists of K (≥ 5) rules where each rule has a relationship among three genes and a class label. Since this second stage of building a classifier is using only the pre-selected genes relevant to the classification, it is capable of increasing classification accuracy while offering affordable computation time even for integrated microarray datasets of large sample size. The experimental results of our system turn out to offer better classification accuracy compared with conventional approaches as the sample size of the training datasets is getting larger. Our two-stage system effectively

Sixth IEEE Symposium on BionInformatics and BioEngineering (BIBE'06)
0-7695-2727-2/06 $20.00

maximizes the use of the accumulated independent microarray datasets, and sheds light on a new paradigm in the field of microarray data integration.

II. Related Works

A. Microarray Data Integration

Until recently, several methods have been proposed for microarray data integration. One of them uses a meta-mining technique [5]. This is a method that integrates and analyzes microarray experiment results individually obtained. However, because the sample size of each individual experiment is generally small, there are many cases where the experimental results themselves are not reliable, and the integration of these results may bring forth an even worse analysis. Another method of integration is to normalize the data obtained from individual research to values having a common scale and then combine them [9]. The most representative example is the case of transforming the data to Z-Scores and combining them. However, this method, which must go through a massive normalization process, costs too much during the preprocessing stage. Studies presenting data integration models besides these include a method [10] that uses Correlation Signature in heterogeneous microarray data integration.

B. Informative Gene Identification

The greatest restraint in analyzing microarray data is that the number of genes is far bigger than the number of samples participating in the experiment [7]. In reality, however, the number of genes that affect classification is very limited, and most genes are noise genes that do not affect class discrimination. Informative genes, as shown in Fig. 2, can be defined as genes showing high expression values on the whole in Class C_1 and low expression values on the whole in Class C_2. On the other hand, genes that do not provide consistent level of expression values for specific classes can be regarded as noise genes that do not have any relevancy [18]. Therefore, it is rational to first identify only the relevant genes that participate in phenotype identification of specific diseases, and then come up with the classification method only using those genes.

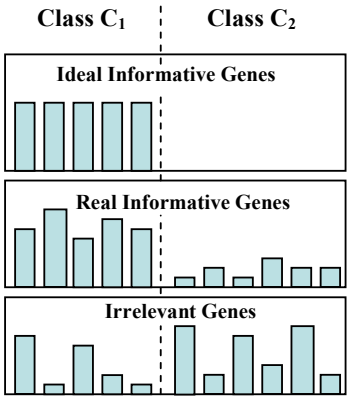

Fig. 2. An informative gene viewed as a type of expression value.

The process that eliminates genes that are not associated with the phenotype of a disease and identifies only the informative genes is called the feature selection, and this is very important to microarray data analysis [2]. Currently, various methods are being presented to precisely and effectively select these informative genes. The linear combination method like the PCA (Principal Component Analysis) [3] is the one of representative methods in feature selection. The PCA method does reduce the dimension of microarray data by using eigen vectors, but it does not individually find genes that are relevant to classification. As another typical feature selection method, the parametric method assumes a statistical model representing the data, like t-statistics or the Golub [8] method, and it saves parameters (ex. mean and variance) that can represent the model. Since this method replaces thousands of gene expression values with very small number of parameters, it has the problem of possibly creating loss of information. On the other hand, the nonparametric method [2][14] lines all sample values of a single gene, and calculates the score (degree of interrupting a complete separation) of how much that gene was differently expressed in the two class groups. When the gene is considered as a feature, the method that is most commonly used among the feature selection methods is the rank-based approach. The rank-based feature selection method measures how much more significant each feature is than the other features in statistical values, and then ranks them and selects the top ranked features. In these approaches, the most popular methods are Information Gain [21], Relief-F [15], and the application using Kendall's Correlation Coefficient [1].

The Information Gain method is an algorithm that uses entropy. Entropy can be defined as the extent of disorder. When X is genes and Y is class label (normal or tumor), entropy formula is as follows.

$$H(Y) = -\sum_{i=1}^{m} pi \log pi$$

$$H(Y \mid X) = -\sum_{x \in X} \sum_{y \in Y} P(x, y) \log P(y \mid x)$$

Then, the Information Gain values (IG) that we want to find are as follows.

$$IG(Y \mid X) = H(Y) - H(Y \mid X)$$

After computing these values for all the genes, the genes that have the high information gain values are qualified as informative genes.

The basic idea of Relief-F is that each gene's weight is calculated by finding the F closest samples (half from the same class (hit), and others from another class (miss)) to each sample. If A is a selected sample and G is a selected gene, weight of the gene is increased in case of hit by the distance between A and hit sample, and decreased in case of miss by the distance between A and missed sample. After performing these computation and aggregation for all, k-genes that have the highest weight are selected.

Park's method [14] builds a binary sequence for a gene, calculates a score measuring how differently the genes are

expressed in the two class groups, by using Kendall's Correlation Coefficient [1]. His method which can be only applicable to a single microarray dataset defines the score as the smallest number of swaps of consecutive digits necessary to arrive at a perfect splitting, with all the 0's on the left and all the 1's on the right. This is shown in Fig. 3.

score	data						Positions swapped
	0	1	1	0	0	1	
+1	0	1	0	1	0	1	3 and 4
+1	0	0	1	1	0	1	2 and 3
+1	0	0	1	0	1	1	4 and 5
+1	0	0	0	1	1	1	3 and 4

Fig. 3. The scoring function in Park's method

All of the methods mentioned above use the expression value of each gene as it is, and there are no consideration regarding the integration and normalization of the microarray data.

C. *Classification*

The most representative of the numerous classification approaches are the SVM [4][19] and the k-Nearest Neighbor [6] methods. The SVM is based on a machine learning algorithm, and it proceeds by learning the linear decision rules, which are represented by hyper planes. The SVM is not only used in microarray classification but also in other various areas, such as regression analysis and density prediction. To apply the SVM to microarray data, because it experimentally needs various types of parameter adjustments, it has the weakness of being fairly complicated. The k-Nearest Neighbor (k-NN) [6] is an algorithm that classifies samples by selecting similar ones from the individual training dataset of the new sample. This k-NN algorithm has the weakness of not providing good efficiency when granting equal weights to all genes.

Among other classification approaches, there are classification methods that do not use parameters but adopt a data-driven machine learning approach, which is called the TSP (Top Scoring Pair) [22], proposed by Xu, and the k-TSP [17] method, proposed by Tan. The TSP is an algorithm that finds a pair of genes with the highest score. For each gene pair, X_i, X_j, the score is the difference of the relative frequencies of occurrences of $X_i < X_j$ in each class. The higher the score, the better the corresponding gene pair discriminates the two classes. Only one pair of genes whose score is the highest is a TSP classifier. The k-TSP classifier extends TSP, and consists of k top-scoring pairs of genes that achieve the best score. For the TSP approach, it is easy to interpret the classifier since just two genes become a single classifier. However, it is plausible that TSP may change even with a small alternation in the training datasets. Also it could be possible that the test sample of independent microarray data does not contain those genes when one wants to predict the class of the test sample. In this regard we are proposing the more reliable classification method which extends the number of genes involved in each rule, and also extends the number of rules in a classifier. TSP and k-TSP proceed to build a classifier without the step of first extracting the informative genes. Since all the genes in the microarray datasets are employed in the classification stage, these methods are computationally expensive as the microarray datasets are getting integrated. In this paper, we start the stage of building a classifier with much reduced, relevant informative genes generated through the first stage.

III. System Overview

The overall system overview is as shown in Fig. 4. In the first stage, the integration of independently generated microarray datasets is being accomplished. Independent microarray dataset has different probe sets and the scale of expression value in each microarray dataset is varied. First of all only the common genes in all microarray datasets are extracted. Then the expression value of each sample in each experiment is transformed into a rank value within the sample. Once the expression values are changed to rank values, the integration of samples originated from different experiments becomes feasible, as long as their gene order is the same. Afterwards, a score that measures how differently a gene is expressed in the two class groups is calculated for each gene. At this stage, genes with a very small score or a very large score could be informative genes. In the second stage, a classifier is built by using only the informative genes that were identified in stage 1. For each set of three genes, X_i, X_j, X_k, one can establish six(3!) magnitude relationships by comparing the rank values of the three genes. For each relationship, among the samples with class C_1 as their label, the number of samples that satisfy the relationship is divided by the number of C_1 samples and saved, and likewise, among the samples with class C_2 as their label, the number of samples that satisfy the relationship is divided by the number of C_2 samples and saved. For every relationship, the difference of these two values is calculated. A relationship with high difference represents a discriminative classification rule. The classifier consists of k classification rules. The parameter, k is determined by applying LOOCV (Leave One Out Cross Validation) to the training dataset. The LOOCV method is an approach that uses

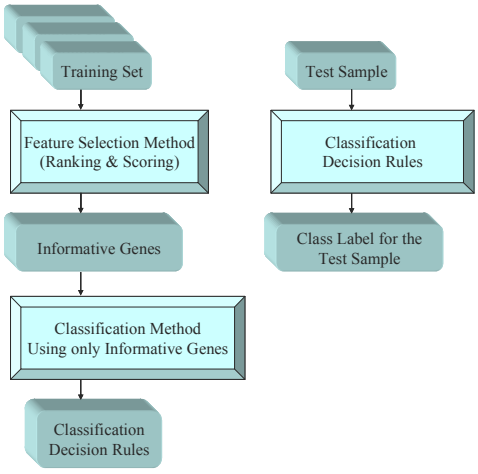

Fig. 4. Overview of our system

all the samples except for one sample in a microarray dataset, builds a classifier, and measures the classifier's accuracy by applying it to the single sample that was excluded. Each classification rule consists of a set of three genes, the magnitude relationship among those three genes, and the prevalent class label of the relationship. Given a new test sample, one can apply the classifier to the sample, predict the class label of the test sample by majority voting, and compare this predicted class with the real class of the test sample.

IV. System Implementation

This section describes the two-stage processing algorithm on the microarray data explained in the previous section. In subsection *A.*, the integration procedure of microarray datasets and informative gene selection algorithm are presented. In subsection *B.*, the k-TST (k Top Scoring Triple) classification method, which compares the magnitude relationship among three genes, converts the relationship into a score, and builds a classifier which consists of k top-scoring relationships is presented.

A. Microarray Data Integration and Informative Gene Identification

As shown in Fig. 5, from the microarray datasets that were generated independently but have the same experimental objectives, only the set of common genes is extracted, as in Fig. 6.

	C_1			C_2		
	S_1	S_2	S_3	S_4	S_5	S_6
G_1	3	5	7	9	11	13
G_2	15	32	23	12	2	3
G_3	33	9	8	11	7	4
G_4	23	4	45	2	22	2
G_5	8	7	7	25	9	9
G_6	9	45	53	43	10	36

,

	C_1			C_2			
	S_1	S_2	S_3	S_4	S_5	S_6	S_7
G_1	0,6	5,7	4,5	5,7	4,3	5,6	8,0
G_2	5,7	7,8	8,9	0,3	0,7	0,5	0,9
G_3	0,5	0,1	0,3	6,9	4,6	8,8	9,9
G_4	7,7	8,8	7,9	0,3	1,0	0,9	0,8
G_5	5,6	6,6	7,7	0,9	1,5	1,2	1,0
G_6	3,4	4,5	3,3	9,9	8,7	8,9	9,0
G_7	0,4	0,2	0,9	9,1	6,7	7,7	5,6
G_8	5,7	4,3	5,6	0,5	4,5	5,6	2,3
G_9	0,2	0,4	0,1	6,7	7,8	8,9	9,9
G_{10}	9,9	8,9	5,6	0,5	0,3	0,5	0,4
G_{11}	0,4	0,5	0,6	7,7	6,7	8,3	6,7

,

	C_1		C_2		
	S_1	S_2	S_3	S_4	S_5
G_3	300	105	183	207	347
G_4	100	205	899	999	678
G_5	560	430	102	203	105
G_6	78	99	890	760	809
G_7	101	201	710	750	800
G_8	500	550	106	210	109
G_9	999	890	210	310	109

Fig. 5. The independently generated microarray data of prostate cancer

	C_1			C_2		
	S_1	S_2	S_3	S_4	S_5	S_6
G_3	33	9	8	11	7	4
G_4	23	4	45	2	22	2
G_5	8	7	7	25	9	9
G_6	9	45	53	43	10	36

,

	C_1			C_2			
	S_1	S_2	S_3	S_4	S_5	S_6	S_7
G_3	0,5	0,1	0,3	6,9	4,6	8,8	9,9
G_4	7,7	8,8	7,9	0,3	1,0	0,9	0,8
G_5	5,6	6,6	7,7	0,9	1,5	1,2	1,0
G_6	3,4	4,5	3,3	9,9	8,7	8,9	9,0

,

	C_1		C_2		
	S_1	S_2	S_3	S_4	S_5
G_3	300	105	183	207	347
G_4	100	205	899	999	678
G_5	560	430	102	203	105
G_6	78	99	890	760	809

Fig. 6. Extraction of the set of common genes among microarray datasets

	C_1			C_2		
	S_1	S_2	S_3	S_4	S_5	S_6
G_3	4	3	2	2	1	2
G_4	3	1	3	1	4	1
G_5	1	2	1	3	2	3
G_6	2	4	4	4	3	4

+

	C_1			C_2			
	S_1	S_2	S_3	S_4	S_5	S_6	S_7
G_3	1	1	1	3	3	3	4
G_4	4	4	4	1	1	1	1
G_5	3	3	3	2	2	2	2
G_6	2	2	2	4	4	4	3

+

	C_1		C_2		
	S_1	S_2	S_3	S_4	S_5
G_3	3	2	2	2	2
G_4	2	3	4	4	3
G_5	4	4	1	1	1
G_6	1	1	3	3	4

Fig. 7. Microarray data expressed as ranks within each sample

Even if the set of common genes has the same order, because of different experimental condition or protocol, the scale of the expression value for each microarray data may be quite different, and a direct integration is inappropriate. We use the rank of expression value for the corresponding gene within each sample rather than using the actual expression value in order to make the direct integration possible. Accordingly, as shown in Fig. 7, the expression values are all converted into ranks within each sample. Our system uses the rank of expression value for the corresponding gene within each sample, sorts the rank levels from the smallest to the largest for each gene along with the class label of each sample which is 0 for normal 1 for tumor, calculates the score which is the number of swaps between neighboring 0 and 1. Table 1 shows an algorithm for identifying informative genes. To help understand the algorithm, let us assume that there is a microarray data as in Table 2 below. Change the data based on the rank within each sample and it becomes Table 3; sort the rank levels from the smallest to the largest for each gene along with the class label of each sample and it becomes Table 4; and change the class label of a sample into binary sequence, and it becomes Table 5.

Table 1. Informative Gene Identification Algorithm

Input:	NI (the number of informative genes), V[][] (expression values)
Output:	IG[][] (Informative genes)
1:	Generate a binary sequence S, which replaces normal samples with 0 and tumor samples with 1.
2:	For all i, j, replace $V[G_i][S_j]$ which represents an expression value, with $R[G_i][S_j]$, which represents the order when they are ranked according to expression values within each sample.
3:	Select an arbitrary gene G_i among genes that were not selected,
4:	For all j, sort $R[G_i][S_j]$ in ascending order, and generate a binary sequence T where normal samples are replaced with 0 and tumor samples are replaced with 1.
5:	Using the scoring function, defined as the number of swaps, calculate the scores for S and T, and insert the score for T into a priority queue with size NI.
6:	Repeat step 3 until there are no unselected genes left.
7:	From the priority queue, select half of NI number of informative genes from top(front), and half of NI number of informative genes from bottom(rear).

Table 2. Data Expressed in Expression Values

	Normal	Normal	Normal	Tumor	Tumor	Tumor
G_1	13	32	3	24	13	42
G_2	25	12	26	3	1	2
G_3	23	6	2	102	59	13
G_4	7	20	63	4	7	27

Table 3. Data Expressed in Rank

	Normal	Normal	Normal	Tumor	Tumor	Tumor
G_1	2	4	2	3	3	4
G_2	4	2	3	1	1	1
G_3	3	1	1	4	4	2
G_4	1	3	4	2	2	3

TABLE 4. AFTER SORTING

	N or T	N or T	N or T	N or T	N or T	N or T
G_1	2 (N)	2 (N)	3 (T)	3 (T)	4 (N)	4 (T)
G_2	1 (T)	1 (T)	1 (T)	2 (N)	3 (N)	4 (N)
G_3	1 (N)	1 (N)	2 (T)	3 (N)	4 (T)	4 (T)
G_4	1 (N)	2 (T)	2 (T)	3 (N)	3 (T)	4 (T)

TABLE 5. DATA EXPRESSED AS BINARY SEQUENCE

G_1	0	0	1	1	0	1
G_2	1	1	1	0	0	0
G_3	0	0	1	0	1	1
G_4	0	1	1	0	1	1

After carrying out the step 1 in Table 1, the initial binary sequence S becomes as "000111" which is a perfect splitting. When we run the function which calculates the score as the number of swaps of consecutive 0s and 1s to arrive at S (perfect splitting) for each gene in Table 5, the gene with the smallest score is proven to be G3 with a total of 1 time, and the gene with the largest score is proven to be G2 with 9 times. This means that G2 and G3 have a strong possibility of becoming informative genes than G1 or G4.

B. k-TST (Top Scoring Triple) Classification Method

In this paper we are making an attempt to generalize the number of genes involved in the rule in order to increase the reliability of classifier for tumor and normal sample prediction. As the first step of this attempt, we are proposing the k-TST.

In k-TST, the number of genes involved in a classification rule is limited to three. For each set of three genes, we establish six magnitude relationships like R_1, R_2, R_3, R_4, R_5, R_6 in the Table 7. For each relationship we calculate the score which is the difference between the probability that the relationship occurs in class 1 and the probability that the relationship occurs in class 2. The set of three genes satisfying the relationship with high score is regarded as most discriminative for classification. Each relationship also keeps its class label by comparing the two probabilities and adopting the class having the prevalent probability. We calculate the scores for all the sets of three genes and for all six magnitude relationships for each set, put the scores into a priority queue in descending order. We use the k top-scoring relationships. Our classifier consists of k classification rules and each classification rule consists of (1) a set of three genes, (2) the magnitude relationship among those three genes, and (3) the class label of the relationship. The algorithm regarding k-TST is in the Table 6. In addition, the scoring function used in step 3 of Table 6 is presented as follows.

TABLE 6. K-TST CLASSIFICATION ALGORITHM

Input:	K (the number of rules specified), IS[][] (informative genes)
Output:	A set of K number of classification rules
1:	From the informative gene set, select a set of three genes that were not processed before.
2:	Determine the magnitude relationships among the three genes for all samples.
3:	Calculate the score for each three gene combination using the scoring function.
4:	Insert the rule which is composed of the calculated score, the gene combination, the magnitude relationship, and the class label of this gene combination into the priority queue with size K.
5:	Repeat the step 1 if there are three gene combinations that are not processed
6:	Select top K rules from the priority queue,

$P_{ijk}(1)$	The probability that a relationship of $X_i < X_j < X_k$ occurs in class 1 (X_i, X_j, X_k stand for the ranks values within a sample)
Δ_{ijk}	$\lvert P_{ijk}(1) - P_{ijk}(2) \rvert$

TABLE 7. K-TST EXAMPLE

	R_1	R_2	R_3	R_4	R_5	R_6	Total
C_1	2	1	29	4	2	4	42
C_2	4	5	1	14	8	1	33

(Definition of R) R_1: $X_i < X_j < X_k$, R_2: $X_i < X_k < X_j$, R_3: $X_j < X_i < X_k$, R_4: $X_j < X_k < X_i$, R_5: $X_k < X_i < X_j$, R_6: $X_k < X_j < X_i$

For example, let us assume that there is a dataset like Table 7. Here, when all the corresponding values of Δ are calculated, in the case of R_3 ($X_j < X_i < X_k$) relationship, one can see that

$$\Delta_{jik} = \lvert P_{jik}(1) - P_{jik}(2) \rvert = \lvert 29/42 - 1/33 \rvert \approx 0.66$$

has the largest score. This means that given a test sample if one observes the R_3 relationship, then the class of the sample is predicted as C_1. Actually we apply k number of classification rules to the test sample, and do majority voting to predict the class of the sample. Majority voting process is as follows.

- *r_i is the i^{th} rule*
- *S is a test sample*
- *k is the number of rules*
- *NC is the number of normal count*

$$L(r_i) = Class\ Label\ of\ the\ r_i\ (L(r_i) = \{normal, tumor\})$$

$$P_{ri}(S) = \begin{cases} L(r_i) & if\ S\ satisfies\ the\ r_i \\ \overline{L(r_i)} & Otherwise \end{cases}$$

$$V(r_i) = \begin{cases} 1 & if\ P_{ri}(S)\ is\ a\ Normal\ Sample \\ 0 & Otherwise \end{cases}$$

$$NC = \sum_{i=1}^{k} V(r_i)$$

If the value of NC is larger than k/2, S is predicted as normal sample, otherwise tumor sample. Since we fixed the number of rules to be an odd number, our system can break the tie and always returns a predicted class label.

V. Experimental Results

In this section, we describe the experiments on the two-stage method in order to verify its accuracy and efficiency. We used prostate cancer microarray data which are publicly available. The platform of these data is Affymatrix HG_95AV2. Each data will be represented as an abbreviation of the first author of the paper, like as Singh [16], Welsh [20] and LaTulippe [11]. Table 8 shows the information about the microarray datasets used in our experiment.

TABLE 8. PROSTATE MICROARRAY DATA

Data	Number of Probes	Number of Normal Samples	Number of Tumor Samples	Total Number of Samples
Singh	12600	50	52	102
Welsh	12626	9	24	33
LaTulippe	12626	3	23	26

A. Determining The Optimal Number Of Rules (k) By LOOCV

In this subsection, we describe the experiment that determines the optimal number of rules (k) by LOOCV. We varied the value of k, and choose the k which gave the highest LOOCV accuracy in each dataset. Since most of the previous gene ranking methods typically select 50-200 top-ranked genes [8][12], we fixed the number of informative genes on 126, 1 % of 12600 genes which is the number of common probe set in the microarray data. We imposed a restriction on LOOCV experiments that k does not exceed 10 and is an odd number in order to break ties in the majority voting procedure. Table 9 shows the summary of the optimal k obtained from the experiments in each training dataset. We measured accuracy, sensitivity and specificity in order to compare our system's performance with others'. They are defined as follows.

$$Accuracy = \frac{The\ Number\ of\ Correctly\ Predicted\ Samples}{The\ Number\ of\ Total\ Samples}$$

$$Sensitivity = \frac{The\ Number\ of\ Correctly\ Predicted\ Tumor\ Samples}{The\ Number\ of\ Tumor\ Samples}$$

$$Specificity = \frac{The\ Number\ of\ Correctly\ Predicted\ Normal\ Samples}{The\ Number\ of\ Normal\ Samples}$$

TABLE 9 THE VALUES OF OPTIMAL K

Training Dataset	Optimal k
Singh	9
LaTulippe	5
Welsh	5
Singh + Welsh	9
Singh + LaTulippe	7
Welsh + LaTulippe	5

In this experiment, the number of rules was restricted to be no less than 5. If the number of rules is too small, the rules can not guarantee the credibility as a classifier. Moreover, there is a possibility that the independent test data might not contain the genes involved in the classifier.

B. Accuracy of Informative Gene Identification Method

In this subsection, we describe the accuracy test of the proposed informative gene identification method. It was compared with Information Gain and Relief-F, which are popular feature filtering methods. Since these two methods cannot be applied to the integrated data directly, we transformed all the data into the normalized form by applying Z-score, which is a classic but most general normalization method. After selecting the informative genes using our proposed method, Information Gain, and Relief-F individually, we compared the LOOCV accuracy of those three gene identification methods. For classification method, we used linear support vector machine (SVM). The classification results of SVM are used to evaluate the effectiveness of our proposed informative gene identification method. Fig. 8 through Fig. 13 show LOOCV accuracy in each dataset.

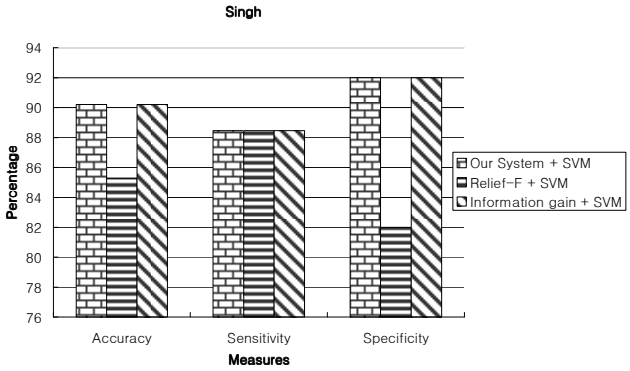

Fig. 8. Accuracy, Sensitivity and Specificity of Singh's LOOCV

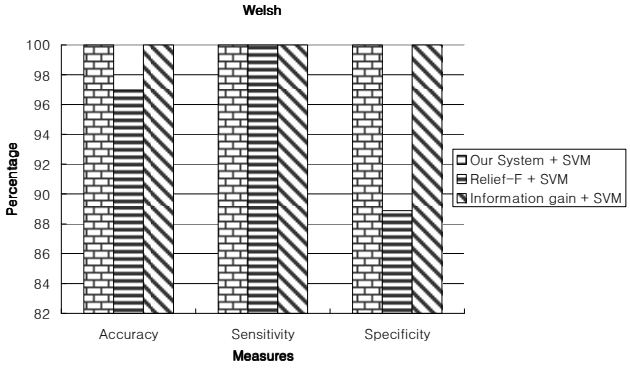

Fig. 9. Accuracy, Sensitivity and Specificity of Welsh's LOOCV

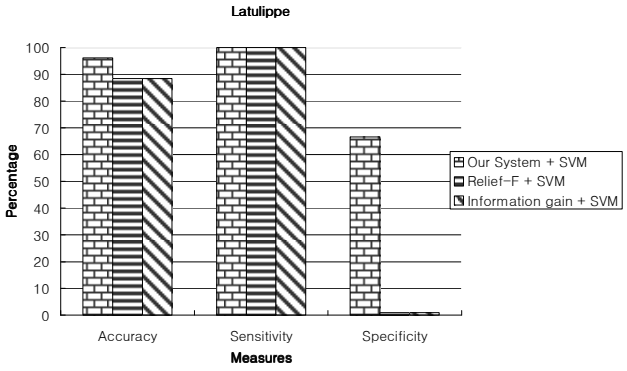

Fig. 10. Accuracy, Sensitivity and Specificity of LaTulippe's LOOCV

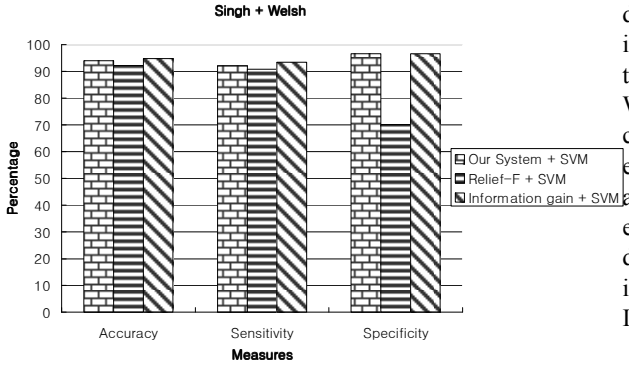

Fig. 11. Accuracy, Sensitivity and Specificity of Singh+Welsh's LOOCV

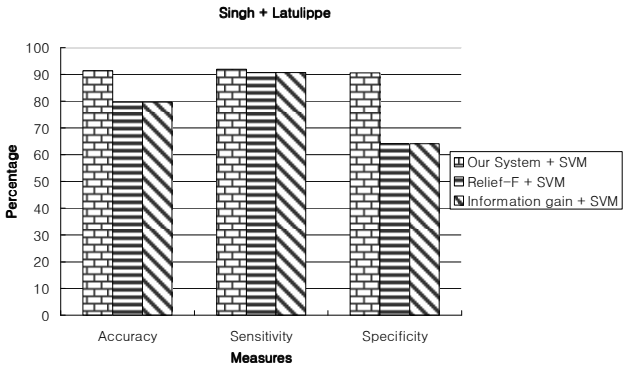

Fig. 12. Accuracy, Sensitivity and Specificity of Singh+LaTulippe's LOOCV

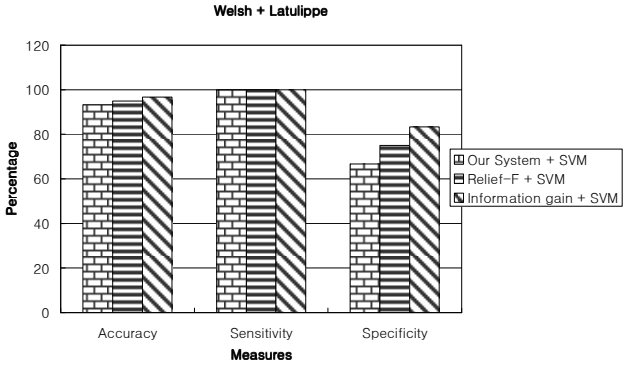

Fig. 13. Accuracy, Sensitivity and Specificity of Welsh+LaTulippe's LOOCV

As one can perceive from the above graphs, the proposed informative gene identification method shows a comparable or better performance than others. Considering LOOCV accuracy on Singh, and Welsh individually, our method shows comparable performance to Information Gain (Fig. 8, Fig. 9). Considering LOOCV accuracy on LaTulippe, our method shows better result. One can see our method always reveals better performance than Relief-F. Especially, when we applied LOOCV to the integrated data of Singh + LaTulippe, better accuracy was obtained by more than 10%. Based on these results, our informative gene identification method offered better (or the same) LOOCV accuracy compared to others.

In Fig. 14 through Fig. 16, the accuracy of independent dataset was presented for Singh, Welsh, and LaTulippe individually. In the case of using Singh as independent test data, training datasets are Welsh, LaTulippe, and Welsh+LaTulippe. We built a classifier from each training dataset, applied the classifier to Singh, and measured the accuracy. The experimental results of accuracy for Singh were compared among different training datasets. We also compared the experimental results of accuracy for Singh among three different methods themselves which are 1) our gene identification method + SVM, 2) Relief-F + SVM, and 3) Information Gain + SVM.

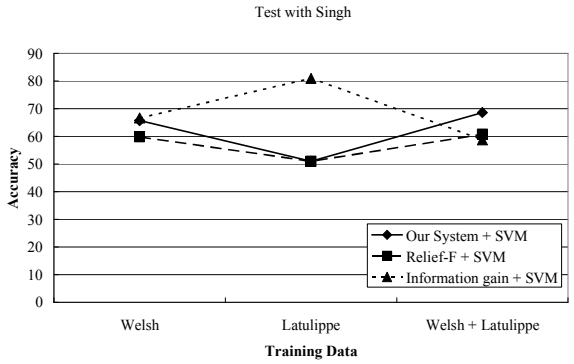

Fig. 14. Accuracy of Singh as test data

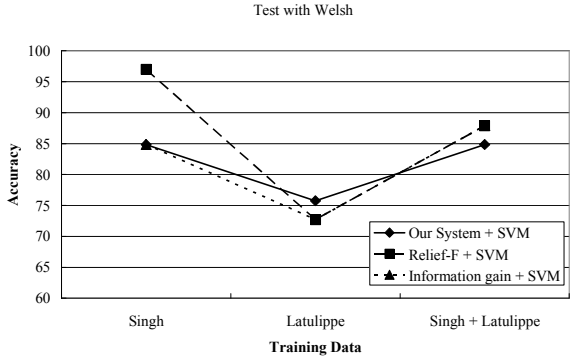

Fig. 15. Accuracy of Welsh as test data

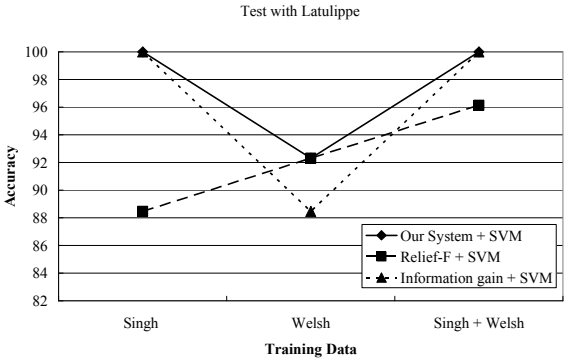

Fig. 16. Accuracy of LaTulippe as test data

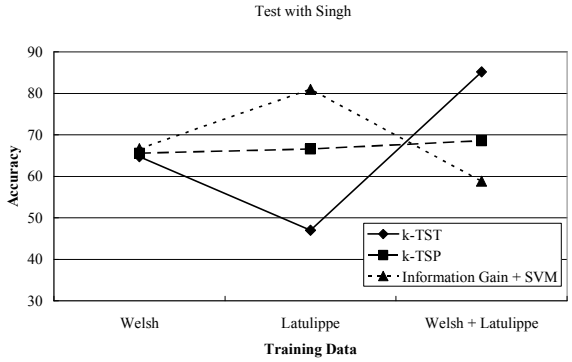

Fig. 17. Accuracy of Singh as test data

The classification accuracy on independent test data also confirmed that the proposed informative gene identification method shows comparable or better performance than Information Gain and Information Gain. In addition, the classification accuracy of our method is getting higher as the sample size in the training datasets is larger by data integration.

C. Accuracy of Classification Method

In this section, we tested the accuracy of our classification method (k-TST) using optimal k acquired from subsection *A.*. We compared our system with SVM after applying Information Gain, which showed a better accuracy than Relief-F and k-TSP.

As shown in Fig. 17 through Fig. 19, we built a classifier using training dataset which are all possible combinations of dataset excluding the test dataset, and measured the accuracy of each independent test data. Table 10 shows the values of optimal k used in our experiments.

TABLE 10. THE VALUES OF OPTIMAL K USED IN OUR EXPERIMENTS

Test data	Training data	k-TSP	k-TST
Singh	Welsh	3	5
	LaTulippe	3	5
	Welsh + LaTulippe	1	5
Welsh	Singh	1	9
	LaTulippe	3	5
	Singh + LaTulippe	5	7
LaTulippe	Singh	1	9
	Welsh	3	5
	Singh + Welsh	9	9

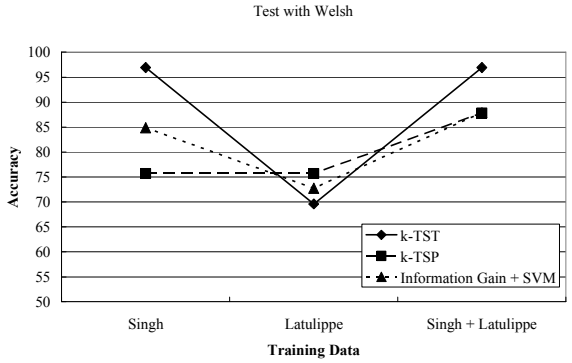

Fig. 18. Accuracy of Welsh as test data

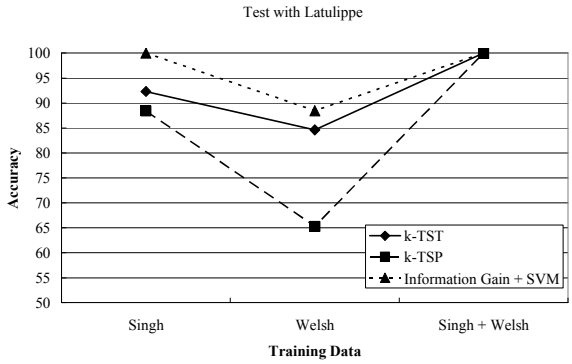

Fig. 19. Accuracy of LaTulippe as test data

As one can see from the above figures, the experimental results of our system did not always show the better performance than other systems when the training dataset is a single microarray dataset. However, it is partly due to the small sample size of single microarray dataset. Especially, both Welsh and LaTulippe consisted of much skewed samples where the number of normal samples is far smaller than that of tumor samples. Therefore, they cannot be used as training data by itself. However, as integration significantly increases the sample size, our system performs much better in accuracy than k-TSP and SVM. Based on these experiments, the proposed two-stage approach can make more credible classifiers than other systems, especially when data are comprehensively integrated.

VI. CONCLUSION

The main contribution of this paper is to introduce a novel two-stage approach which sequentially combines integrating independent microarray datasets and selecting informative genes, and builds a classifier. With an abundant supply of publicly available microarray data, a new method which integrates independently-generated microarray data with the same experimental objectives was employed in the first stage of selecting informative genes. Increasing sample size by integrating the independent microarray data has lead to the discovery of more reliable classifier. Moreover, two-stage approach makes the computation time of the second stage tremendously lessen because only the pre-selected informative genes were considered. Since the number of genes involved in our classifier is relatively small, they could be very cost-effective in a clinical setting where microarrays with thousands of genes are impractical. A prototype was implemented and tested on integrated prostate microarray datasets. Experiments show that our method of informative gene selection is better than or comparable to other methods. Being compared with Information Gain plus SVM, our system also shows better classification accuracy on independent test data as the sample size is getting larger by integration. We haven't tested the multi-class classification yet. Currently the classifier of this system can be extended to multi-class classification using pair-wise ensemble without problems. For identification of informative genes in multi-class environment, we can still build a perfectly split sequence, and calculate the score to reach the sequence. However this needs some careful considerations if there are some correlations among classes. Also we are currently investigating (1) cross platform validation where cDNA data is included in the integrated microarray data, (2) elimination of redundancy among informative genes and (3) generalization of the number of rules and the number of genes involved in a rule in our classifier.

REFERENCES

[1] N. Bailey, "Statistical methods in biology," Cambridge university press, 1995.

[2] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and Z. Yakhini, "Tissue classification with gene expression profiles," Journal of Computational Biology, vol. 7, pp. 559-583, 2000.

[3] C. Bishop, "Neural networks for pattern recognition," Oxford University Press, New York, 1995.

[4] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machine," 2001. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[5] J. K. Choi, U. Yu, S. Kim, and O. J. Yoo, "Combining multiple microarray studies and modeling interstudy variation," Bioinformatics, vol. 19, pp. 84-90, 2003.

[6] B. Dasarathy, "Nearest Neighbor Norms: NN Pattern Classification Techniques," IEEE Computer Society Press, Los Alamitos, CA, 1991.

[7] S. Dudoit and J. Fridlyand, "Classication in microarray experiments," Statistical Analysis of Gene Expression Microarray Data, Chapman and Hall, 2003.

[8] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Collier, M. L. Loh, J. R. Downing, and M. A. Caligiuri, "Molecular classification of Cancer: class discovery and class prediction by gene expression monitoring," Science, vol. 286, pp. 531-537, 1999.

[9] H. Jiang, Y. Deng, H.S. Chen, L. Tao, Q. Sha, and J. Chen, "Joint analysis of two microarray gene-expression data sets to select lung adenocarcinoma marker genes," BMC Bioinformatics, vol. 5, pp. 81-92, 2004.

[10] J. Kang, J. Yang, W. Xu, and P. Chopra, "Integrating heterogeneous microarray data sources using correlation signatures," In International Workshop on Data Integration in the Life Sciences (DILS), 2005.

[11] E. LaTulippe, J. Satagopan, A. Smith, H. Scher, P. Scardino, and V. Reuter, "Comprehensive gene expression analysis of prostate Cancer reveals distinct transcriptional programs associated with metastatic disease," Cancer Research, vol. 62, pp. 4499-4506, 2002.

[12] L. Li, W. Leping, 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

[13] D.R. Rhodes, T.R. Barrette, M.A.Rubin, D. Ghosh, and A.M. Chinnaiyan, "Meta-Analysis of microarrays : interstudy validation of gene expression profiles reveals pathway dysregulation in prostate Cancer," Cancer Research, vol. 62, pp. 4427-4433, 2002.

[14] P. J. Park, M. Pagano, and M. Bonetti, "A nonparametric scoring algorithm for identifying informative genes from microarray data," Pacific Symposium on Biocomputing, pp. 52-63, 2001.

[15] M. Robnik-Sikonja and I. Kononenko, "Theoretical and empirical analysis of relief f and rrelieff," Machine Learning, vol. 53, pp.23-69, 2003.

[16] D. Singh, P. G. Febbo, K. Ross,D. G. Jackson, J. Manola, and C. Ladd, "Gene expression correlates of clinical prostate Cancer behavior," Cancer Cell, vol. 1, pp. 203-209, 2002.

[17] A. Tan, D. Naiman, L. Xu, R. Winslow, and D. Geman. "Simple decision rules for classifying human Cancers from gene expression profiles," Bioinformatics, vol. 21, pp. 3896-3904, 2005.

[18] C. Tang, A. Zhang, and J. Pei, "Mining Phenotypes and Informative Genes from Gene Expression Data," ACM SIGKDD, pp. 24-27, Washington DC, 2003.

[19] V. Vapnik, "Statistical Learning Theory," John Wiley & Sons, New York, 1999.

[20] J. B. Welsh, L. M. Sapinoso, A. I. Su, S. G. Kern, J. Wang-Rodriguez, and C. A. Moskaluk, "Analysis of gene expression identifies candidate markers and pharmacological targets in prostate Cancer," Cancer Research, vol. 61, pp. 5974-5978, 2001.

[21] I. H. Witten and E. Frank, "Data mining: practical machine learning tools and techniques with Java implementations," Morgan Kaufmann, 1999.

[22] L. Xu, A. Tan, D. Naiman, D. Geman, and R. Winslow, "Robust prostate Cancer marker genes emerge from direct integration of inter-study microarray data," Bioinformatics, vol. 21, pp. 3905-3911, 2005.

0-7695-2727-2/06 $20.00